(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A FRAMEWORK FOR CLASSIFYING AND RESOLVING SEMANTIC CONFLICTS USING THE ENHANCED ENTITY-RELATIONSHIP MODEL

by

Daniel A. Lindsey

September 1992

Thesis Advisor: Magdi N. Kamel

Approved for public release; distribution is unlimited.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DCLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL *(If Applicable)* 55	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS *(city, state, and ZIP code)* Monterey, CA 93943-5000	7b. ADDRESS *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a. 6b. OFFICE SYMBOL *(If Applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS *(city, state, and ZIP code)*	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*
A FRAMEWORK FOR CLASSIFYING AND RESOLVING SEMANTIC CONFLICTS USING THE ENHANCED ENTITY-RELATIONSHIP MODEL

12. PERSONAL AUTHOR(S)
Lindsey, Daniel A..

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT *(year, month, day)*	15. PAGE COUNT
Master's Thesis	FROM TO	Sept. 1992	138

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUBGROUP

18. SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*
Data Modelling, Entity Relationship Diagrams, Database Integration Semantic Conflict, Heterogeneous Databases

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*
In today's organizations, information in current databases is stored in a variety of heterogeneous systems and data organizations. This situation causes problems when trying to integrate them into a federated or multidatabase solution. Particularly troublesome is semantic conflict, or differences in the meanings of data structures and definitions in heterogeneous databases. This thesis proposes a systematic approach towards identifying, classifying and resolving semantic conflicts. Using an entity relationship approach as the integrating model, a framework is developed which describes all possible semantic conflicts among the underlying schemas. This framework can be employed as a methodological tool during an integration effort. Possible resolution strategies are offered for each type of conflict and applied to the separate databases to realize a common global schema which could be used to formulate effective queries against the total original volume of data.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT [X] UNCLASSIFIED/UNLIMITED [ ] SAME AS RPT. [ ] DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Magdi N. Kamel	22b. TELEPHONE (Include Area Code) (408) 646-2494 22c. OFFICE SYMBOL Code AS/KA

DD FORM 1473, 84 MAR   83 APR edition may be used until exhausted   SECURITY CLASSIFICATION OF THIS PAGE

All other editions are obsolete   Unclassified

Approved for public release; distribution is unlimited.

A Framework for Classifying and Resolving Semantic Conflicts Using the Enhanced Entity-Relationship Model

by

Daniel A. Lindsey
Lieutenant, United States Navy
B.A., Oregon State University, 1984

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September, 1992

Author: ______________________
Daniel A. Lindsey

Approved by: ______________________
Magdi N. Kamel, Thesis Advisor

____
Kishore Sengupta, Second Reader

____
David R. Whipple, Chairman
Department of Administrative Sciences

ABSTRACT

In today's organizations, information in current databases is stored in a variety of heterogeneous systems and data organizations. This situation causes problems when trying to integrate them into a federated or multidatabase solution. Particularly troublesome is semantic conflict, or differences in the meanings of data structures and definitions in heterogeneous databases. This thesis proposes a systematic approach towards identifying, classifying and resolving semantic conflicts. Using an entity relationship approach as the integrating model, a framework is developed which describes all possible semantic conflicts among the underlying schemas. This framework can be employed as a methodological tool during an integration effort. Possible resolution strategies are offered for each type of conflict and applied to the separate databases to realize a common global schema which could be used to formulate effective queries against the total original volume of data.

TABLE OF CONTENTS

I. INTRODUCTION

A. BACKGROUND

Organizations have over time developed many disparate databases to manage information. These databases have been implemented using a wide variety of incompatible models, languages and storage methods. Migration of database systems to an integrated strategic Information Resource based architecture will require the interoperability of these diverse sources of data. Conflicts among these heterogeneous databases will impede consolidation efforts.

This problem can be addressed in the near term by transforming the schemas of incompatible data organizations, such as hierarchical and relational, into a common data model which will capture all information contained in the original databases and make it available to the user in a unified form. The Enhanced Entity-Relationship model, which is both semantically rich and conceptually simple, can serve as an integrating model for combining the data from different databases.

With the independent databases represented in equivalent schemas, a framework for the identification, classification, and resolution of semantic data conflicts can be developed. The integrated global schema can guide the formulation of queries, and the detailed understanding of semantic conflicts among the component databases resulting from the re-engineering process be incorporated in the design of a global controller which can manage the retrieval of information from a federated database application. Users requiring access to data from several disjoint databases can then process queries against the reconciled common schema.

In the longer term, data element standardization efforts may obviate many of the semantic conflicts addressed by this thesis. However, different preferred forms of organizing corporate information will remain specific to various functional domains. Tools for the integration of data from heterogeneous databases will still be required.

B. OBJECTIVES

The objective of this thesis is to develop a framework for identifying, classifying, and resolving semantic conflicts using the Enhanced Entity-Relationship model. This includes transforming heterogeneous databases into a common schema for comparison and identification of semantic conflicts, illustrating all possible forms of semantic conflict, both at the schema and at the data level. Using real world examples, the classification framework will be applied to diverse database applications in the course of an integration effort. Finally, this analysis will suggest, in general terms, resolutions to the various semantic conflicts identified through the use of the framework. Integration of the component databases into a global schema and design guidelines for the implementation of a global controller completes the objectives.

C. RESEARCH QUESTIONS

The following research questions are addressed in this thesis:

(1) What qualities are needed in an integrating data model to integrate data from multiple sources?

(2) What types of semantic data conflicts arise in heterogeneous databases, and what is an appropriate framework for classifying semantic conflicts?

(3) How can semantic data conflicts best be identified and resolved to allow integrated access to corporate information stored in databases using different data models, definitions, and constraints?

(4) How might semantic conflicts be resolved to allow the formation of a common global schema incorporating heterogeneous databases which use different data models, definitions, and constraints, and what guidance can the re-engineering process give toward the design of a global controller component for a federated database application?

D. SCOPE AND LIMITATIONS

This thesis will review the potential semantic data conflicts which can arise in heterogeneous databases, and develop a framework for classifying these conflicts. A common data model for use in integrating diverse sources of data is examined and evaluated for appropriate qualities. General measures to resolve these conflicts with the aim of integrating the data and useful in the design of a functional federated database controller will be examined.

This thesis will not address heterogeneity at the platform or Database Management System level. Real world database specifications from Department of Defense users will be analyzed at the level of descriptive detail obtained. Reasonable assumptions will be made (in terms of detailed data definitions, etc.), where necessary, to illustrate the types of semantic conflict under discussion.

E. METHODOLOGY

The methodologies used in this thesis include:

(1) Obtaining database specifications for several real world applications from the same functional domain (i.e., administrative personnel management).

(2) Examining an appropriate common integrating conceptual data model for integration of diverse databases.

(3) Transforming the separate databases into equivalent schemas, using the conceptual integrating data model.

(4) Analyzing and comparing the equivalent schemas to develop a framework for identifying and classifying all possible semantic conflicts.

(5) Exploring possible solutions to the conflicts identified, and using the framework and resolution heuristics to integrate a global schema which subsumes all available data from the candidate databases.

(6) Using the knowledge of semantic conflicts gained to suggest design strategies for a global controller component to manage a federated database including the subject databases.

(7) Reviewing the experience of the integration process to suggest future areas of research into useful techniques for resolving semantic heterogeneity.

F. ORGANIZATION

The thesis is organized in the following manner:

Chapter II addresses the proliferation of heterogeneous databases in organizations. This includes an analysis of various levels of heterogeneity, and suggests sources of different kinds of conflicts.

Chapter III reviews the required qualities of a suitable integrating model, with particular mention of the various types of existing databases which might have to be modeled. The common conceptual model used in this thesis is explained, and examples are given of the diagrammatic conventions used in following chapters.

Chapter IV presents a real world scenario of heterogeneous databases drawing on specifications obtained from various Department of Defense functional applications. The federated database approach to integration is described, including the role of the global controller component which manages the resolution of semantic conflicts at the functional level. Each database is transformed into a common equivalent schema using the integrating model.

In Chapter V, the equivalent schemas developed in Chapter VI are systematically compared to form a classification framework of semantic heterogeneity. Examples of each type of semantic conflict are illustrated and discussed based on the specifications detailed in the appendices.

Chapter VI explores in general terms possible means of resolving each type of semantic conflict expressed in the classification framework. The proposed solutions are then applied to the individual schemas to create a common global schema which includes all information originally available. Additionally, this chapter applies the semantic conflict framework to theoretical design considerations of a federated database approach to integration.

The concluding chapter reviews lessons learned in the course of integrating real world heterogeneous databases, and offers conclusions about identifying and resolving semantic conflicts between databases. Recommendations and suggested areas of future research are offered based on the results of this analysis.

II. PROLIFERATION OF HETEROGENEOUS DATABASES IN ORGANIZATIONS

A. HETEROGENEITY IN DATABASES

In a perfect world, the advantages of interoperability would motivate end users, designers, and developers to ensure that seamless and effective information sharing were built into database applications from the ground up. Still, **heterogeneous databases** have proliferated throughout organizations for a variety of reasons. A (largely) homogeneous paradigm would be practical for an organization entering the database field from a standing start, with access to the full spectrum of currently available technology. Gradual evolution, however, has resulted in the current situation. Organizations such as DoD have continuously developed database applications over 40 years. Recurring cycles of hardware, software, and technology during that time have all contributed to the diversity of databases in use today. In addition to these essentially technical issues, the incremental, disjoint, and arbitrary implementation of conceptual design methodologies has contributed to the present chaotic assortment of incompatible systems.

This evolution has resulted in a database environment with three levels of heterogeneity. At the lowest level, different database applications are implemented on a wide range of hardware platforms. Similar hardware can run a variety of operating systems. Distributed databases must communicate, using compatible communications protocols. Variation in these protocols introduces more conflict. At the next level, Data Base Management Systems

(DBMSs) may be incompatible, even when intended to work with similar data structures. Finally, when data is named, defined and organized into a particular architecture, subjective design choices introduce fundamental and potentially intractable semantic conflicts. Platform and DBMS heterogeneities are discussed in section one, while semantic heterogeneity is discussed in section two.

This thesis deals with semantic conflicts which arise at the schema, or architectural level of database organization, and at the data level. Such conflict arises from both technical and methodological causes. Incidental heterogeneity issues, caused by hardware, operating system, DBMS software, and communications protocol variations are not addressed.

1. Platform and DBMS Heterogeneity

When information was first stored for electronic manipulation by computer systems independent from the specific application programs doing the manipulation, it was organized as 'flat files'. These were simple, identically formatted records, accessed by the application program through the program's explicit knowledge of where in the record a given fact could be found. No attempt was made to make associations between individual records, since each was treated by the application as a unique piece of information. Additions to the set of records was therefore easy, but a change in the structure of a record very difficult, since the entire application had to be rewritten to preserve the necessary explicit internal map of the record's structure.

Initial interest in database research centered on the management of data in business applications such as automated payroll, inventory, and transaction processing. These domains required efficiency in accessing and modifying very large amounts of data, and were oriented toward well defined, repetitive processes which could be run from start to completion in a batch mode. Additionally, these first databases appeared when the physical limitations of the available hardware imposed very definite restrictions on the architecture which could be used to organize the data.

These factors influenced the hierarchical, or tree-based approach to data management. Data records are assembled into a collection of trees, some being root records, and all others having a unique parent record. This organization is amenable to the simple relationships of employee to wage, tax code, dependents, etc. in a payroll scenario, or the assembly to subassemblies to parts relation of an inventory. Since the processing is repetitive, and need not be done in real time, hierarchic database programs can be optimized to navigate through the tree structure even when this is highly complex. Finally, the hierarchic data model was suited to magnetic tape storage, an economic requirement before random access disk-based storage became affordable.

Evolutionary modification of the hierarchic data architecture led to the Conference on Data Systems Languages (CODASYL) standard. This arrangement allows more complex, and thus more useful, relationships between data elements to be represented, with records arranged into a directed graph or network. Efficient implementation of the network organization both required and exploited the more flexible capabilities of direct access storage media. Disks rapidly replaced tape as their cost-performance ratio improved. Application programming for a network database requires a highly procedural navigation oriented language, like the hierarchic model, which restricts the degree of dynamic processing changes available to the end user.

The relational data model was pioneered in the early nineteen seventies and offered a fundamentally different approach to data storage. Data is represented as simple tabular structures (relations), and access is allowed through a high level, non procedural query language. The complexity of relationships between data elements is unrestricted. The application programmer, or end user, specifies a predicate which identifies the desired record or combination of records, and the DBMS translates that specification into an efficient algorithm which performs the database access.

Even the most advanced relational models are not without drawbacks, however. The computational complexity of solving queries involving multiple large relations can be prohibitive, and much research has gone into the optimization of relational queries. Efficient design, or normalization, of the relations themselves to eliminate redundancy and logical anomalies has also required theoretical advances. New approaches to allocation and management of disk space and memory buffering routines have been necessary to minimize storage cost and access delays. While the relational data model provides the maximum flexibility in organizing and manipulating data in the early nineties, it does so at some cost.

The evolution of theoretical work on data storage and processing, and the hardware development which facilitated and paralleled it represent the technical factors which lead to heterogeneity in databases. As applications were developed and brought into production, organizational pressures prohibited continuous re-engineering of applications to exploit each new theoretical or hardware development, even where that was appropriate. It must be kept in mind that some degree of heterogeneity in organizational databases is not an aberration which can, or even should, be completely eliminated. A relational database offers the flexibility to deal with arbitrarily complex, unstructured queries on an ad hoc basis, but its computational overhead does not recommend it for a mature inventory system. When update processing requirements are relatively static and well understood, transactions against the inventory can be done periodically, and the number of records is very large, a hierarchic database is a better choice. Over the range of organizational activities within the DoD, various problem domain solutions will naturally fall to diverse appropriate data model/hardware combinations.

2. Semantic Heterogeneity

A separate causal factor leads to semantic data conflicts. Generally, these arise from variations in database design methodology and implementation. The technical factors discussed above concern physical application level strategies and models. The hierarchic, network and relational data architectures deal with how individual data elements are organized, physically linked, retrieved, and manipulated by the hardware and software of an application. As suggested, technical issues lead to a natural, unavoidable diversity in organizational databases, based primarily on processing efficiency within particular problem domains. Methodological factors, on the other hand, result from human individuality, differences in perception, and preferences. They give rise to heterogeneity between databases addressing the same functional application, using identical hardware, operating systems, data models, and DBMS software. Because data definition, naming conventions, and conceptual organization are inherently subjective issues, semantic conflicts are almost guaranteed amongst databases developed by different teams in the absence of strictly enforced strategic design guidelines.

Individuals interpret the world from their own personal perspective. Organizations, and subdivisions of organizations, have similarly diverse views of their environment. Items of interest, which become data elements, aggregations of elements, which become records and logical entities in organizational databases, are named, defined, and organized in this qualitative, subjective, environment. If two departments of the same company undertook to develop personnel databases, without specific guidance from the front office, it would not be surprising to find different names for similar employee attributes, identical field definitions for contradictory elements, or even completely different ways of structuring the problem. This is a predictable and unsurprising consequence of individual and organizational differences. It is germane that the type of conflict described could, and would, arise even if central guidance was provided, but was restricted to mandating a particular hardware/DBMS suite.

Yet this is exactly how many organizations, including the DoD, have developed their database applications over the last forty years. Until very recently, only particular hardware, operating systems, or DBMSs have been standardized among the services and their various departments. There was still no strategic guidance which provided common definitions, naming conventions, etc. at the element or entity level. Thus even if DBMS/platform conflicts do not arise, semantic conflicts remain which can make databases incompatible.

B. THE NEED FOR INTEGRATION OF HETEROGENEOUS DATABASES

As organizations mature in the use of information technology, the potential benefits of consolidating heterogeneous databases become irresistible. Vital corporate information is captured, stored, and available to decision makers and operational functions from many database applications, but incompatibilities can prevent the integration of data from different sources. Elimination of data redundancy, to achieve cost advantages, means more applications must share compatible data. Data accuracy, critical for high-risk decisions, can be enhanced by identifying disjoint data among similar databases and resolving the semantic conflicts.

The need for standardization of data management has been recognized by the DoD and forms a central part of the Corporate Information Management (CIM) initiatives. Current data dictionary efforts, which address the problem of semantic data discrepancies at their lowest level, hold promise for ameliorating the problem in future applications. There is also an urgent need for high level methods to allow the integration of information in currently existing heterogeneous databases.

Two approaches have been identified which address this issue. The multidatabase approach leaves the component databases in their native form, but provides transparent access to all included information. Users are aware that they are dealing with multiple diverse databases, both schematically, and physically. Alternatively, the federated approach consolidates the component databases under a global schema, and gives both location and heterogeneity transparency. Users interact with the data as though it were in a single, physically contiguous, logically consistent database. Either approach requires a strong logical data model to describe multiple individual physical data architectures. The next chapter addresses a suitable integrating model.

Once all databases of concern have been expressed in a common conceptual schema, semantic conflicts among individual data elements can be easily identified. Chapter IV presents three heterogeneous real-world databases and describes the process of transforming them into equivalent schemas in the common integrating model. Chapter V develops a framework of semantic heterogeneity for the integrating model. The framework enables the classification of semantic data conflicts stemming from human variation in methodological implementation. With a comprehensive integrating model, and a taxonomy for identifying semantic heterogeneity which includes, schematic and data conflicts, possible solutions can be proposed. This is the subject of Chapter VI. Conflicts in architecture and data organization which arise at the Platform/DBMS level are properly addressed by the detailed implementation of the integration effort. The resulting consolidated, reconciled information can be accessed through appropriate systems to provide organization-wide use of existing heterogeneous databases.

III. THE ENHANCED ENTITY RELATIONSHIP MODEL

A. DATA MODELING

When a database application is developed, the segment of the real world to be modeled is analyzed in light of the users' requirements. The designers make a choice about the conceptual data model to be used. The choice of model is governed by its perceived appropriateness to the problem domain, the personal preference of the designers and their familiarity with various methodologies. Conceptual modeling is done at a very general level of analysis, and has only marginal impact on implementation decisions. The data elements and arrangements suggested by the analysis must then be formally specified, and their structure and behavior defined in terms of the logical model. The ultimate physical organization of data (in a network, relations, etc.) is independent of the logical schema used for design, and is chosen as a function of processing, access requirements, transaction frequency, and the structure of the resulting schema.

In considering heterogeneous databases with a view toward information sharing and consolidating access, the original logical design is often unavailable, and the conceptual model used unknown. The final application architecture may provide no indication of the conceptual scheme used in the initial analysis. A logical integrating model which can describe multiple diverse implementation models is needed to subsume the heterogeneous component databases and allow them to be expressed in a consistent schema.

Of the potential candidates for an integrating conceptual model, the Entity-Relationship (ER) approach stands out as a strong candidate. It is semantically rich, conceptually simple, and can capture arbitrarily complicated relationships between atomic elements and larger groupings of information. It is widely used in database design [Ref. 1], and offers a natural and intuitively understandable way of displaying information and real world relationships. With the additional semantic expressiveness provided by extensions to the ER model (referred to as Enhanced Entity Relation, or EER), newly popular concepts such as inheritance can also be defined.

Although sophisticated renditions of EER schema become diagrammatically complex, the essential representation of atomic data elements as connected attributes which describe an entity, or real world item of interest, is fundamental. Relationships between entities, and the characteristics of the relations (cardinality, mandatory participation, etc.), are explicitly defined and represented by the model, making it simple to visually interpret an ER schema. The ER/EER data model is one of the most widely used logical schemes for conceptual database design [Ref. 2]. This wide acceptance, as well as its superior descriptive qualities, make it the most appropriate integrating model.

1. Top Down Modeling

In top down database modeling, the user's real world, or the portion of interest, is analyzed in terms of data requirements and relationships. Appropriate data types are defined, and the information is arranged in logical groupings which meet the users' needs. At this level of modeling, no implementation details are considered, and the resulting schema is easy to understand and verify with non-technical users. The basic tool for this process is the conceptual data model.

For example, consider the design of a database to organize information about officer personnel for the Department of the Navy. The user has specified that the information of interest includes basic data, such as name, rank and serial number, as well as the officer's duty assignment. The designer, using the EER technique, takes these requirements, and arranges a conceptual schema which represents the officer as an entity, defined by the attributes of name, rank, and social security number. Likewise, the unit he or she is assigned to is shown as an entity, defined by a unit identification code attribute. The relationship between the officer and the unit is also represented.

The user also provides specifications about appropriate data types for various elements. Name might be most usefully defined as a character string, while rank is desired to be represented by some arbitrary code which fits into the user's overall information processing philosophy. At this point, uniquely defining, or key, attributes are defined for entities where possible confusion could exist between two sets of information. This could occur if two officers had identical names and ranks. When the conceptual schema is complete, the user confirms that the information and arrangement meets the database requirements, and implementation proceeds through the mapping of the conceptual schema to a DBMS, and design of physical data storage structures.

2. Bottom Up Modeling

The use of conceptual design techniques such as the EER model in a bottom up manner differs in that the purpose is not to capture a suitable schema from real world information. Instead, the intent is to reverse engineer a conceptual schema from an existing database implementation. Data types, file structures and attribute definitions have already been designed and implemented. Transforming the low level database implementation specifications back into a high level conceptual schema allows analysis of the choices made in arranging the original data requirements.

More important, bottom up data modeling can render completely different database implementations in equivalent form for comparison and interpretation. This is the main thrust of reverse conceptual modeling in this thesis. The EER model is semantically rich enough to conceptually represent many existing database implementation. The EER model will be used in Chapter IV as a common model to transform diverse heterogeneous databases into equivalent schema's, for analysis of potential semantic conflicts. The following sections present the EER concepts and the diagrammatic conventions used in this thesis. The specific EER model used throughout this thesis is taken from [Ref. 3]. The closing section of this chapter briefly describes the application of the EER modeling concepts to bottom up analysis of various different database implementation types.

B. EER CONCEPTS

The Enhanced Entity Relationship model is essentially very simple. Information is represented by entities, which are described by attributes, and associated to each other by various kinds of relationships. With intuitive extensions of these three simple concepts, arbitrarily complex views of the real world can be expressed in a graphic and easily understood way. Since the use of the model here is not to capture a top down schema from beginning user requirements, not all of the semantic expressiveness available will be described. For a complete examination, the reader is directed to [Ref. 3].

1. Entities

The central object in the EER model is the entity, which represents a real world 'thing' with independent existence. It may be something with physical existence, such as an officer, or a concept, like a security clearance. Entities are described by properties, which are real world facts about an entity. They are also associated with other entities to capture additional information.

Entities can be unique, and independently defined, or they can be dependent on the existence of another entity. Such entities are referred to as 'weak'. A security clearance entity is an example of a weak entity, since in the real world, it doesn't make sense to think of that entity without a related officer, who holds the clearance. Weak entities have their own attributes, and represent important real world concepts, but must be associated with an identifying owner to have meaningful semantic content.

Figure 1 illustrates an entity type.

Figure 1. Entity Type

Figure 2 is a weak entity type, having no useful semantic content without an identifying relationship.

Figure 2. Weak Entity Type

2. Attributes

Descriptive facts about entities are called attributes. They can be simple, single valued attributes, such as a social security number, or they might be multivalued, or even made up of other attributes. Composite attributes make it possible to represent data which may be handled as a whole sometimes, but in part at others. An officer's name might be a composite attribute, if it is used in full (Last, First, Middle) in some instances, and sometimes in part (Last only).

A critical attribute concept is that of the key. A key attribute is one which uniquely defines the entity it describes. This allows distinguishing between instances of an entity type for which all other attributes are identical. Social security number is a very common key attribute. A related concept is that of the partial key. A partial key attribute uniquely describes a weak entity when concatenated with the key of the weak entity's identifying owner.

A final very useful attribute type, is the derived attribute. This represents information which is not explicitly captured in the database, but may be determined, or calculated, from related information. The total number of officers assigned to a unit, for instance, could be calculated from the number of related officer entities for each instance of the unit entity. Total number assigned could then be assigned to the unit entity as a derived attribute.

Figure 3 shows how a simple attribute is depicted graphically.

Figure 3. Attribute

Figure 4 depicts a key attribute, and a partial key attribute.

Key Attribute Partial Key Attribute

Figure 4. Key Attribute and Partial Key Attribute

Figure 5 illustrates a multivalued attribute (an attribute with a single meaning, for which an entity might have multiple instances).

Figure 5. Multivalued Attribute

Figure 6 represents a composite attribute.

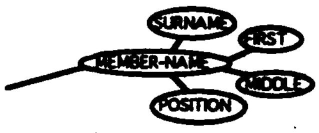


Figure 6. Composite Attribute

Figure 7 shows how a derived attribute is diagrammed.

Figure 7. Derived Attribute

Figure 8 shows a partial completed entity with its descriptive attributes.

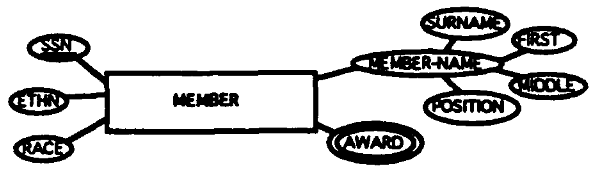


Figure 8. Member Entity

Figure 9 represents a weak entity with its partial key.

Figure 9. SCRTY-CLEAR Weak Entity

3. Relationships

The third basic concept in EER modeling is the relationship. This is used to represent associations of varying types between entities. An officer, for example, could be related to a unit by the relationship 'Assigned To'. The completed schema then makes it explicit that an officer is assigned to a unit, by connecting the two entities with a relationship. Weak entities are associated with their identifying owners by an identifying relationship.

Relationships can capture a very large range of semantic meaning by the addition of relationship cardinality. Cardinality refers to constraints on the relationship. In other words, if every officer is assigned to one and only one unit, this is defined in the EER schema by adding a cardinality number to the relationship in the direction from the officer to the unit. Units, logically, would have many officers assigned, and this would be represented by an appropriate cardinality in the relationship direction from the unit to the officer. The graphical conventions used to depict cardinality will be shown in the following section, and their usage will be more obvious.

Additional constraints on relationships are referred to as partial and total participation. This can be visualized by considering the weak entity example above. Since a security clearance has no semantic meaning without an identifying relation to an owner officer entity, security clearance participation in that relation must be total. In other words, each and every instance of security clearance must participate in the identifying relationship with some officer entity, or it cannot exist in the schema. Conversely, it is possible to conceptualize a unit, perhaps newly formed, which has no officers assigned. This allowable meaning is represented by a partial participation relationship. A unit entity is allowed to exist without necessarily participating in a relationship with a particular officer. By combining participation and cardinality constraints on relationships, any conceivable association of entities can be modeled using EER techniques.

Figure 10 is an example of a simple relationship.

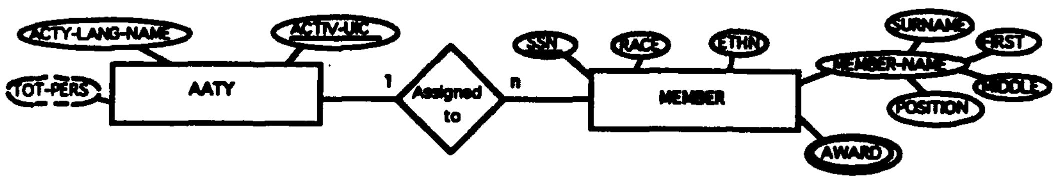


Figure 10. Relationship of AATY to MEMBER

Figure 11 illustrates the identifying relationship between a weak entity and it's identifying owner. The double diamond around the relationship specifies that it is an identifying relationship. The double line connecting the weak entity to the relationship is used to indicate the total participation of the weak entity. This is a condition of the identifying relationship, but not restricted to this relationship type. Any relationship type can be constrained on either side by total participation.

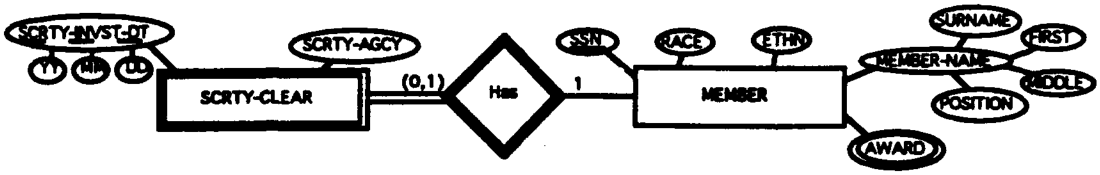


Figure 11. Identifying Relationship of SCRTY-CLEAR to MEMBER

Figure 12 gives examples of various cardinality constraints on relationships. The cardinalities are read in the direction away from the constrained entity. In other words, for the cardinality label immediately above the connecting line, each entity is related to one and only one instance of the other entity (one-to-one). Next above shows the ENTITY 1 related to many ENTITY 2 (one-to-many) Finally, above the line, is an example of many entities on either side related to many entities (many-to-many).

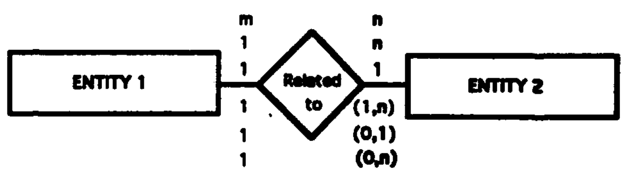


Figure 12. Various Cardinalities of Relationships

Below the line are illustrated more complex cardinality constraints. These are read identically, in terms of direction, the conceptual extension being the range defined in the parentheses. The left number of the ordered pair represents the lower bound on participation, and the right number the upper bound. Thus, reading the example immediately below the line depicts a relationship in which the ENTITY 1 must be related to at least one, and may be related to any number of ENTITY 2 Conversely, ENTITY 2 is related to one and only one ENTITY 1. The remaining cardinality constraints are read in a similar manner, and are not exhaustive.

4. Complex Data Organizations

Complex and useful data organizations such as those becoming popular in Object Oriented analysis are represented in the EER model by specialized extensions to relationships. Two will be described below. The Generalization/Specialization structure, which captures the concept of inheritance, and the aggregation structure, which captures the whole-part relationship concept.

a. Generalization/Specialization

The Generalization/Specialization (Gen/Spec) relationship is used to model a schema of entities, which all posses common attributes as part of their description, but which for some subset of entity instances, unique attributes define logical subclasses. It is sometimes referred to as an IS-A(N) relationship (i.e., the Specialization entity IS-A Generalization entity). A simple illustration which expands on those used above is to consider a personnel database containing data not only on officers, but all members of a given service. For all entities representing service members, a large number of attributes, such as name, social security number, etc., will be same. That is, all members will possess these attributes. Officer members, however, will have different attributes than enlisted members, and it is conceptually elegant to be able to model this phenomenon explicitly.

This is done with the Gen/Spec relationship which connects the generalized entity member, to the specialized entities enlisted, and officer. Thus for a given instance of officer, the full set of defining attributes consists of those belonging to the generalized member, in addition to the specific subset of attributes which define the specialized officer. The officer instance 'inherits' the attributes of its related member instance.

Gen/Spec relationships can be extended with various qualifications, just as simpler relationships. Two Gen/Spec constraints utilized in this thesis are those of total participation of the general entity, and disjointness. Total participation represents the semantic concept that each and every member of instance of the general entity must belong to one or more of the related specialization entity types. If on the other hand, it were allowable for a general entity to exist independently (that is, only possess the generalized attributes), the general entity's participation in the Gen/Spec relationship would be partial.

Disjointness indicates that each specialization entity must belong to only one specialization. In the member to enlisted/officer relationship, disjointness is enforced, since each member must be either an officer, or an enlisted. Alternatively, a Gen/Spec relationship in which a specialization entity could belong to more than one specialization would be an overlapping type.

Figure 13 diagrams a disjoint Gen/Spec relationship with the constraint that each and every MEMBER must belong to either the MEMBER-OFR, or the MEMBER-ENL specialization. Disjointness is represented by the small 'd' in the relationship circle, and total participation of MEMBER by the double line connecting MEMBER to the relationship.

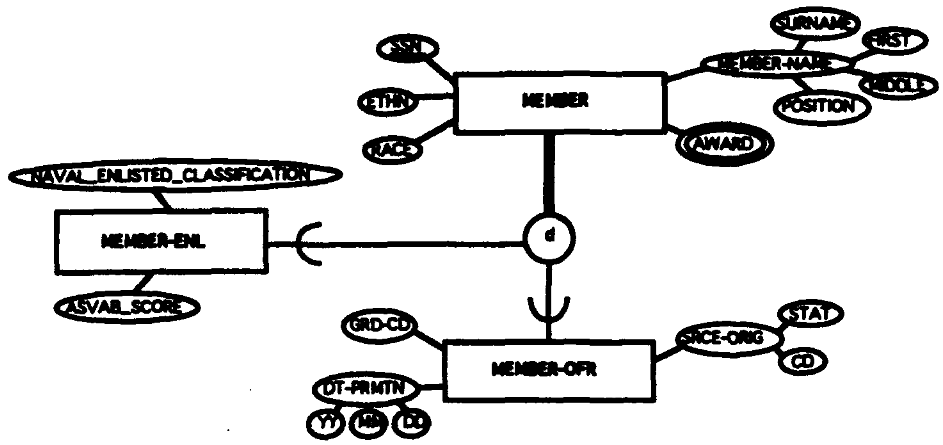


Figure 13. MEMBER to MEMBER-OFR Generalization/Specialization Relationship

b. *Aggregation*

Aggregation is an abstraction concept for building composite entities from component entities. This can be thought of as a PART-OF relationship. For instance, Army, is PART-OF Department of Defense. This is extremely useful in EER modeling of some advanced database applications.

C. APPLICATION TO REAL WORLD DATABASES

This section reviews, in general terms, some of the conceptual bottom up modeling techniques applicable to transforming existing database implementations into equivalent EER schemas. In some ways, reverse conceptualization of existing data organizations is simpler than top down design, since many of the structural choices have already been made, and may be obvious. On the other hand, absent the original design, some arbitrariness is inevitable, and it falls to the re-engineering analyst to make logical choices. While randomness of data structuring is a danger of this approach, if consistent criteria are used, the purpose of the modeling, which is to allow methodical comparison of semantic schemas, will be fulfilled.

1. Flat Files

Conceptually, a flat file data organization can be simply rendered as a single entity possessing all the attributes defined in its record structure. However, this approach is inelegant, and loses much of the semantic content which is likely represented by the original file record. Thus, repeating groups of fields are sought, and extracted as separate entities with appropriate relationships to the entity suggested by the major category of the record. Similarly, if a record has fields which are utilized for different meanings depending on the values of other fields, this suggests that the record actually describes a Gen/Spec organization, and is so mapped to the EER diagram. Accurately recreating the cardinalities of relationships is the most difficult part of bottom up modeling, since these constraints, while explicitly represented by the EER schema, are in general enforced at the implementation level, and often are not included as part of the available database definition. Additionally, whether or not a particular entity is 'weak', and the choice of identifying owner for those which are, may not be obvious. In these cases, the re-engineering analyst must make logical and consistent choices based upon knowledge of the information domain.

2. Hierarchic

Basic, restricted data organizations such as the hierarchic, or tree based structure, can be fully described quite simply. The relationship between entities is one-to-one/many, and the presence of specific attributes may allow the collection of several elements into attributes describing a generalized entity. Layered grouping of entities as 'children' of other entities requires no extension of the concept.

3. Network

The network model builds on the tree architecture, but allows additional associations between entities. This arrangement is restricted by the condition that a 'child' may have only one 'parent' of a given type. The resulting multiple one-to-many relationships form the network, for which the model is named. Like mapping the hierarchic structure to an EER schema, no particularly complicated analysis is necessary, other than choosing appropriate cardinality of relationships, if this is not expressly defined for the re-engineering analyst.

4. Relational

Relational data structures are not constrained in the complexity of connections between data elements and entities. But in modeling the relational implementation to an EER schema, most often one may proceed from the assumption of a correspondence between tables and entities Relationships are explicitly given by the distribution of foreign keys. Cardinalities may have to be inferred, as in the previous cases.

5. Object Oriented

Although not widely available, there is great interest in commercial database implementations which exploit the use of object oriented design. The generalization, aggregation, and inheritance constructs offered by the EER model are powerful and desirable data ordering concepts. In the future, integration of heterogeneous databases including Object Oriented implementations will use these descriptive properties of the EER model more fully than they are employed here.

IV. HETEROGENEOUS DATABASE SCENARIO

A. THREE OFFICER PERSONNEL ADMINISTRATIVE DATABASES

Personnel information for the Department of Defense is currently stored in a variety of separate and diverse databases. A great wealth of data is available, but is maintained by different organizations, using different database management systems (DBMS), design philosophies, and hardware platforms. Frequently, data that span across several databases need to be retrieved. Under the current environment, however, integrating the total information presents many difficulties.

To accomplish the integration, the many conflicts arising between the multiple databases must be resolved, to allow global querying of the body of data. Platform incompatibility, such as that between diverse Operating Systems, manufacturers' physical hardware implementations, etc., is solvable, although sometimes at great cost in processing resources. At the level of DBMS heterogeneity, programming techniques can be used to translate a query appropriate to a relational database into one suitable for searching a flat file structure. At the semantic level, conflicts of meaning, and conceptual arrangement of data must be reconciled.

Of the three levels of heterogeneity, semantic conflicts are the most difficult to resolve. During the conceptual design of a database application, the meaning and structural organization imposed on real world information of interest fundamentally influences every subsequent use of that stored data.

Even when identical DBMSs, platforms, operating systems, etc., are considered, many conflicts can still arise due to the different meanings assigned to the same real world item by different designers.

This chapter examines three actual administrative databases currently in use by various organizations within the Department of Defense to maintain information on commissioned officer personnel. DBMS and platform differences amongst these databases will be ignored except where these issues influence the effort to identify, and classify, semantic conflicts.

1. Active Duty Military Inventory (ADMI)

The ADMI database is maintained by the Defense Manpower Data Center, and includes data on all active duty military personnel, both officer and enlisted. It is a tape-based flat file database, and serves primarily to process batch transactions for various reports of interest to the manpower office of the Secretary of Defense. Information on Naval commissioned officers is therefore available as a subset of the records of the ADMI database. A partial database specification for ADMI is presented in Appendix A. While not complete, in terms of complete data definition, the level of detail available is representative of what might actually be available during the course of integrating a multidatabase application. Reasonable assumptions have been made as to exact attribute definitions, in some cases to illustrate a particular point of potential semantic conflict.

The ADMI database stores basic information of interest to the personnel administration function, such as name, rank, social security number, and date of birth, sex, race, etc. It also keeps data on marital status, number of dependents, and whether a member's spouse is also a member of the military. In addition to these facts, the ADMI database contains an extensive number of statistical elements concerning a member's status on original entry to military service. This includes height, weight, test form number, both raw and adjusted scores for the Armed Services Vocational Aptitude Battery (ASVAB), and place of entry into the service.

2. Officer Personnel Information System (OPINS)

The OPINS database is maintained by the Bureau of Naval Personnel to track commissioned officer assignment, promotion, and qualification status. Like the ADMI, it is a flat file database, and it theoretically contains the entire population of interest for this scenario. A partial specification is presented in Appendix B. Similar assumptions as to attribute definitions have been made, but in both cases, attribute names have been taken directly from the specification as listed in the appendix.

OPINS stores similar common personnel information to that in the ADMI database, such as name, rank, sex, etc.. The data reflects important differences in the OPINS area of interest, however. It contains relatively detailed data about an officer's educational history, both civilian and military, as well as the military qualifications resulting from that training. The officer's promotion status and history is captured very explicitly, including year group, precedence number, and the dates of accession to each rank. The unit assignment data in OPINS differs from the brief essentials kept by the ADMI database in being far more extensive. Historical assignments, by billet number, primary and collateral duty, dates assigned, and projected rotation date for the current assignment are maintained for each officer.

3. Inactive Manpower and Personnel Management Information System (IMAPMIS)

The IMAPMIS database is maintained by the Naval Reserve Force as an integrated repository of information on all members of the Naval Reserve. This includes both officer and enlisted reserve personnel, as well as active duty Naval personnel in the Training and Administration of Reserves (TAR) field. It is a relational database and is the most recently implemented of the three. Partial table definition for IMAPMIS is presented in Appendix C. Fewer assumptions at the attribute level were required in analyzing IMAPMIS, as the available definition is far more complete than for ADMI or OPINS.

Like the ADMI database, and the OPINS, the IMAPMIS maintains the essential administrative data needed by the personnel function (name, rank, pay entry base date, etc.). It also stores a wide variety of unique information specific to the Naval Reserve manpower management process. This includes reserve unit affiliation, in addition to mobilization unit assignment, last paid drill, total credited drills, whether drills were voluntary or mandatory, and retirement points accumulated. The training data captured by the IMAPMIS is also the most extensive of the three systems, including the information available in both the OPINS and the ADMI databases, as well as data elements indicating reserve officer training accomplished by enlisted members, reserve mobilization training evolutions, and service experience in military operations. As was seen in the variation of informational content between the ADMI and OPINS databases, the specific facts recorded in the IMAPMIS reflect the different area of interest of its users.

In all cases, the assigned definitions are intended to be realistic, and consistent with the design of the database in question. The assumed definitions should not be taken as representative of any actual data definition in use for the given database, and are only presented for the purpose of illustration.

B. SOURCES OF HETEROGENEITY

1. Database Management System / Platform

It is obvious that the three (ADMI, OPINS, and IMAPMIS) have different implementation details. While ADMI and OPINS may in fact run under identical DBMSs, hardware, and operating system, IMAPMIS certainly runs under an incompatible DBMS, and has a different hardware/operating system combination. Any heterogeneity this situation may or may not introduce to the multidatabase scenario under discussion is not germane to this analysis. The focus of this analysis is the effort to identify and resolve the semantic conflicts which are present.

2. Semantic

Since the three databases under discussion were all developed and implemented at different times, by different organizations, for different purposes, it should not be surprising that very different conceptual arrangements have resulted. A review of the Defense Manpower Data Center ADMI database reveals a very different area of interest, for instance, than that of OPINS. The Defense Manpower Data Center is concerned with issues such as total military end-strength, allotment of personnel resources to budgetary program elements, and the like. OPINS, on the other hand, being a service-specific database, captures a very different set of data for a given officer, including present and past assignments by billet, and promotion year group. There is a large overlap in the area of basic information (name, rank, SSN, etc.), but it is obvious that the designers of OPINS were interested in a different view of the commissioned officer than that presented by ADMI. IMAPMIS data overlaps both ADMI and OPINS, and additionally captures information of specific interest to the personnel management of the Reserve force, such as Reserve unit affiliation, and last credited drill period.

Besides varying areas of informational interest, the three database design efforts employ very different naming conventions. ADMI largely employs plain language labels for data elements which are easily understood. OPINS uses much more service-specific language, which would be obvious to someone familiar with Navy terminology, but perhaps confusing to a layman. IMAPMIS follows the OPINS terminology closely, but since it is described in a particular DBMS language, the entity and attribute names are awkward and not always easily matched to their corresponding elements in ADMI and OPINS. This results in a great deal of semantic heterogeneity, since it becomes an important issue to resolve whether each designer means the same thing when an attribute is called UNIT, for instance.

C. ATTEMPTING TO QUERY THE TOTAL BODY OF DATA

Information on the population of interest, Naval Commissioned Officers, is contained across all three databases. Frequently, queries that span the three databases need to be answered. For example, we may like to retrieve all available data for a given value of a key attribute, such as Social Security Number. Obviously, a query against any one of the databases cannot ensure this. Information on all officers may not exist in a single database. For instance, an active duty officer not in the TAR program will not appear in

IMAPMIS. As was pointed out, different attributes representing real world items of interest are contained in different databases.

To guarantee no loss of any information already available, we must somehow present a global query which will be processed against a global schema that represents the integration of the three databases, and return the requested information. Even when this is accomplished, the further problem of conflicting data remains. In other words, due to differences in update times, data entry errors, etc., even identical attributes for the same officer may contain different data values.

Therefore, because of the different data organizations, naming conventions, and particular information available in each database, as well as the situation where conflicting data represents the same information, there must be some means of resolving the inevitable semantic conflicts which will arise when particular attributes are returned.

D. INTEGRATION STRATEGY

To allow queries that span several databases, a federated database approach is suggested. Following this approach, each local database is considered a logical component in the federation. These components are tied together by a global schema that represents the integration of the local schemas. To accomplish this several steps are necessary. First, each local schema is transformed into an equivalent schema in a semantically rich common data model. This step is carried out in the following sections using the Extended Entity Relation (EER) model, applying the concepts and diagrammatic conventions covered in Chapter III. Second, a systematic comparison is made across the individual equivalent schemas between corresponding entities, and attributes, searching for potential conflicts. Third, after resolving semantic conflicts, the local schemas in the common data models are merged to form a global schema. Fourth, an additional control component, known as the global controller, is required. The global controller maintains the definition of the global schema and acts as a coordinator and translator: it receives a global query, possibly in a user specific language; translates it into an equivalent query on a common-model global schema; decomposes and translates the common-model query into subqueries to the corresponding local database sites for processing; collects the results; identifies and resolves data content conflicts; reformats the result; and sends it back to the originating site. The first three steps of this process are covered in detail in the remainder of this thesis. The theoretical design of the query and resolution components of the global controller described in step four, above, is related to the levels of schematic and data heterogeneity covered by this analysis. Chapter VI will show how the methods of semantic conflict resolution developed can be applied to the design of the global controller. The specific implementation of the global controller deals largely with the levels of DBMS and platform heterogeneity mentioned earlier, and is outside the scope of this study.

Due to the large number of attributes comprising the real world sample, this analysis extracts a representative subset of attributes from each database. This subset adequately illustrates the methodology employed. Similar treatment of the complete ADMI, OPINS, and IMAPMIS schemas would follow identical procedures.

The remainder of this chapter deals with transforming the ADMI, OPINS and IMAPMIS schemas into equivalent EER schemas. Chapter V uses these diagrams to identify a comprehensive set of potential semantic conflicts among equivalent EER schemas using examples from the three databases. Chapter VI employs this classification framework to suggest potential solutions for each type of conflict and complete the realization of a comprehensive global schema.

1. Translation of ADMI Into EER Form

Deriving an EER diagram from the ADMI database was begun by selecting an appropriate subset of attributes from the total which comprise each ADMI record. The specific attributes were chosen to ensure that similar information was analyzed from each database, as well as to realistically show the differences in domains of interest. Once the set of data elements was determined, they were grouped as attributes of a logical arrangement of real world entities. These entities were then related based on a reasonable interpretation of the conceptual view which ADMI is attempting to represent.

Since ADMI is a flat file, all data elements it contains can in some sense be considered simple attributes of a single entity. However, certain analytical standards are applicable. The repeating set of fields used to represent LANGUAGE, for instance, clearly represents a multi-valued composite attribute which is appropriately diagrammed as a separate entity. Since the ADMI database contains information on all active duty personnel, fields which take on different values depending on officer/enlisted status, and specific service membership, can be diagrammed as defining attributes of Generalization/ Specialization relationships. This is how the relationship of active duty member to service member to specific service officer is modeled. The shaded entities for other service member, and naval enlisted, are included in the diagram only to indicate the structure of the relationship, and are not populated with the describing attributes they would possess in a complete representation. In the actual implementation of ADMI, there would not be a separate instance of the UNIT entity, since it is merely a set of attributes of the member record. In reverse engineering from a flat file database to an EER, however, it is proper to represent UNIT as an entity, having existence independent of its relation to a particular member. In this way, the most general level of conceptualization is achieved. This is analogous to the convention which would be followed in modeling the real world top down to an EER schema. The particular relation of unit and member in the actual ADMI is only an artifact of a given implementation decision.

Obviously, some of the results of the flat file to EER translation shown below are based on arbitrary assumptions, and may be open to challenge. The process detailed here is representative of what would be done in a more rigorous manner if, for instance, the multidatabase designer had access to information on the intentions of the designers of the original database. At the conceptual level of this treatment, the effort is to illustrate the procedure, and ensure that all the various potential conflicts are enumerated. While detailed translation of the ADMI might result in a slightly different EER diagram, it is not felt that any undue artificiality has been introduced into the example.

The entity structure extracted from the ADMI database is presented in Appendix D. The completed EER diagram of the extracted attribute subset is shown in Figure 14.

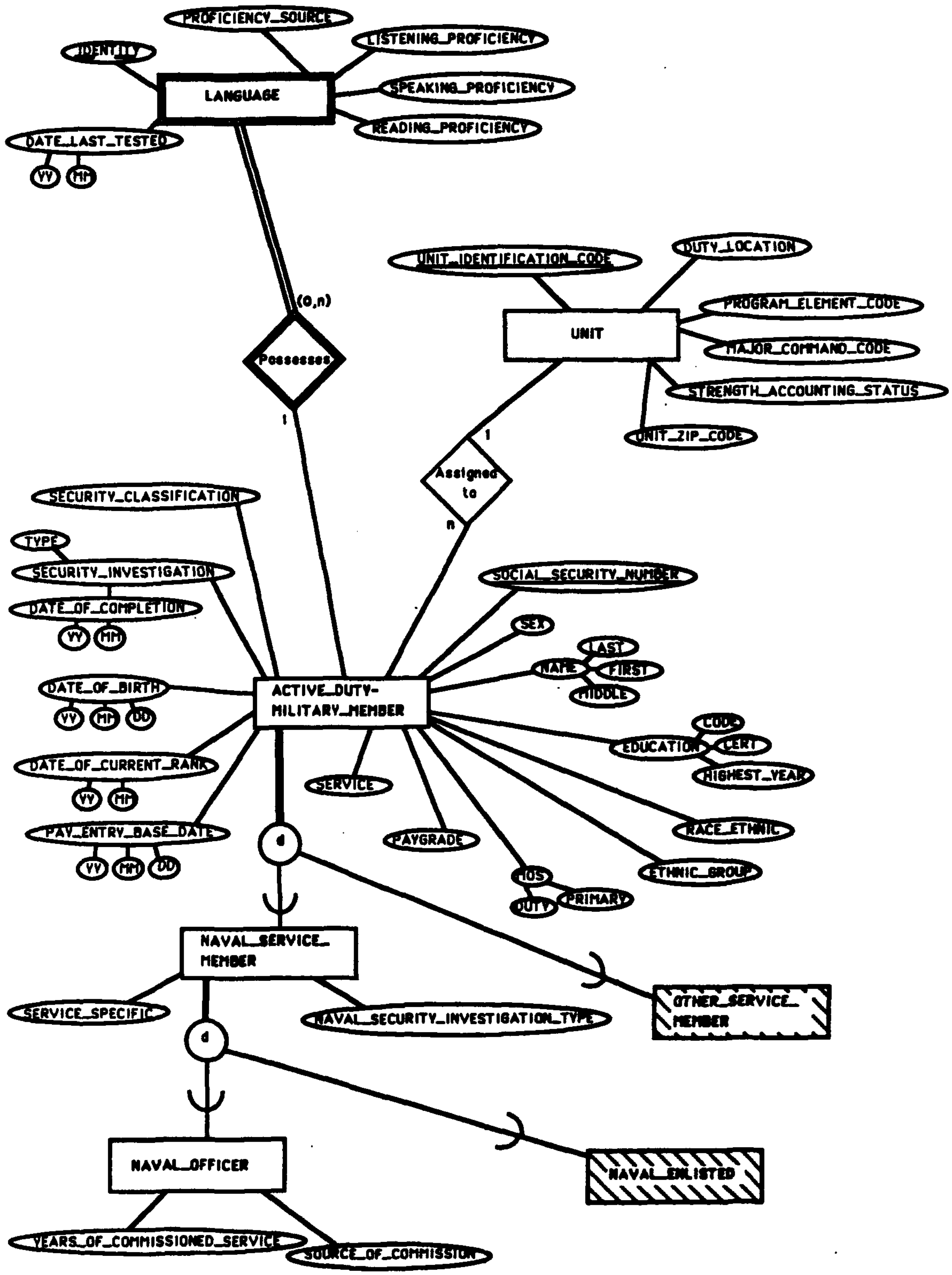


Figure 14. EER Schema for the Active Duty Military Inventory

2. Translation of OPINS Into EER Form

An identical translation process was performed on the OPINS flat file database. The extracted subset in this case resulted in a substantially different EER diagram, though very similar attributes were utilized. This points out the semantic differences which arise in each designer's representation of the real world. The entity UNIT, for instance, is derived from a repeating composite attribute in the OPINS record, and is diagrammed as a separate entity having a one-to-many relationship with COMMISSIONED_OFFICER. This is different from the relationship between MEMBER and UNIT in the ADMI example, because OPINS actually captures a historical record of unit assignments, vice simply the current one. Likewise, the entity YEAR_GROUP has no matching construct in ADMI, since this represents information of interest solely to the designers of OPINS.

Similar caveat is offered regarding the exact process of translation for OPINS as was true for ADMI. No claim is made for the fidelity of the EER diagram as translated, relative to the actual real world view intended by the OPINS designers. However the results given here are representative of the use of the EER process and model to formulate a bottom up conceptual schema from an existing database.

The entity structure extracted from the OPINS database is presented in Appendix E. The completed EER diagram for OPINS attribute subset is shown as Figure 15.

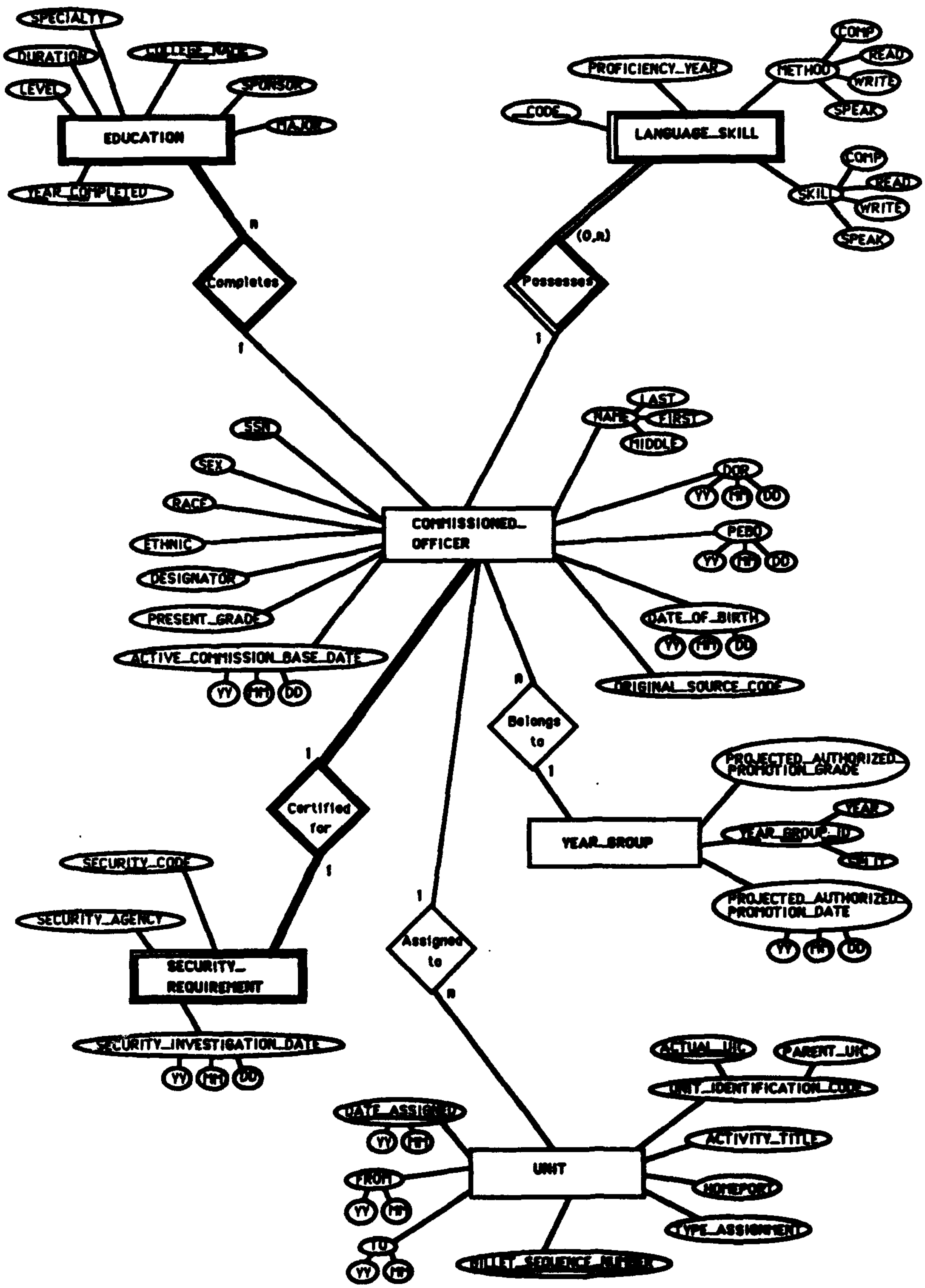


Figure 15. EER Schema for the Officer Personnel Information System

3. Translation of IMAPMIS Into EER Form

Unlike the potentially arbitrary assumptions required in translating the ADMI and OPINS flat files to EER form, the conversion of IMAPMIS is more straightforward. Since IMAPMIS is a relational database, in most cases there is a simple correspondence between the IMAPMIS tables as defined, and the entities modeled. Some entities, such as LANG, are not specified uniquely as separate tables by the IMAPMIS specifications, though they are referred to as individual record types. Relationships for the IMAPMIS EER diagram are easily derived from the location of foreign keys within the tables.

The entity structure extracted from the IMAPMIS database is presented in Appendix F. The EER diagram for the IMAPMIS subset as translated is shown as Figure 16. The shaded entity for enlisted member is included as a place holder only to indicate the structure of the relationship, and is not populated with the describing attributes it would possess in a complete representation.

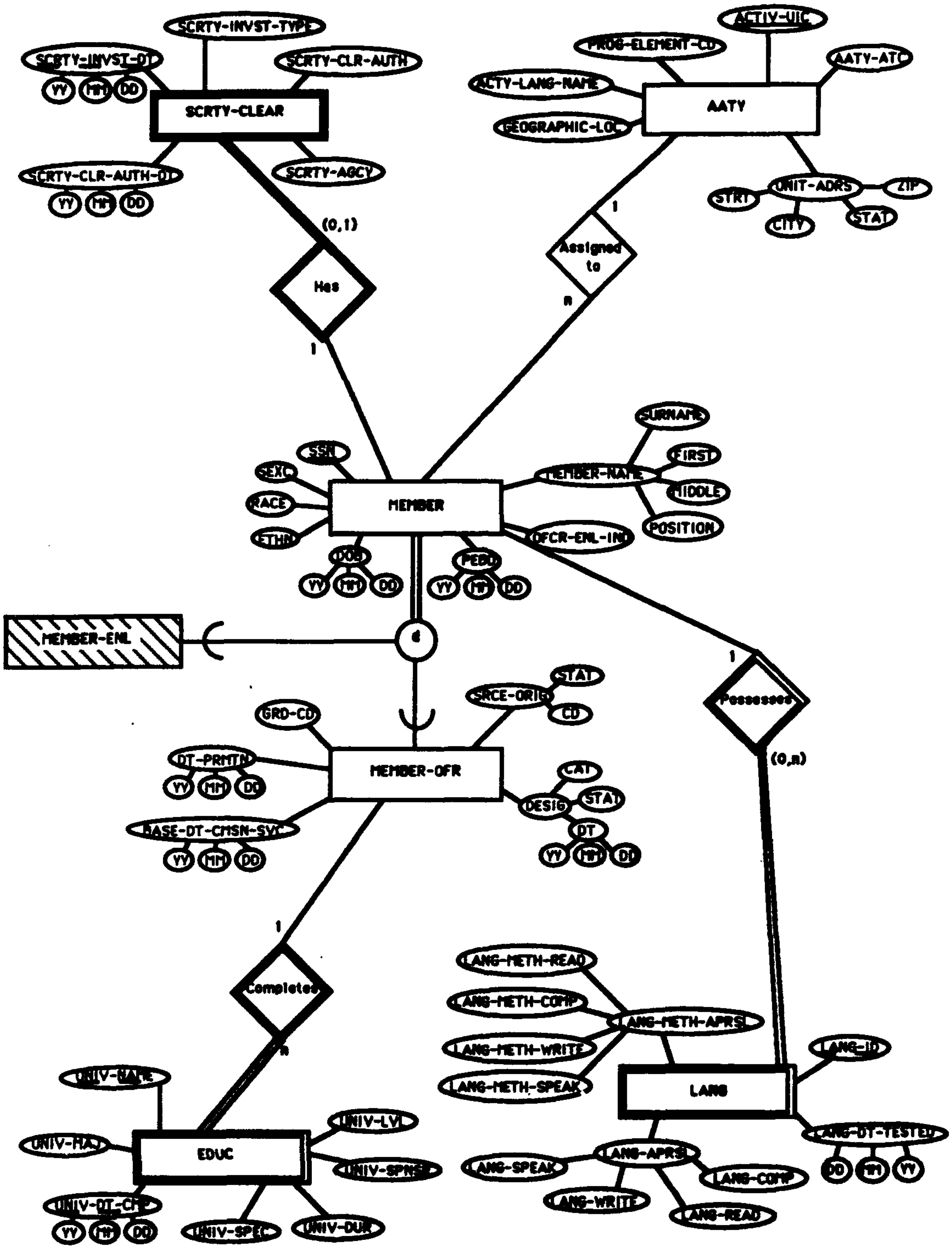


Figure 16. EER Schema for the Inactive Manpower and Personnel Management Information System

V. A FRAMEWORK FOR SEMANTIC HETEROGENEITY

A. CLASSIFYING SEMANTIC CONFLICTS

With the candidate databases transformed into equivalent EER schemas, potential semantic conflicts can be identified. To facilitate the classification and resolution of semantic conflicts, a framework for identifying such conflicts is developed in this chapter. The classification framework presented here recognizes two broad kinds of conflict. Schematic conflict, which occurs at the level of the conceptual organization and definition of the database, and data level conflicts, which occur between the actual data values returned from the different databases by a query against the global schema.

Procedurally, the individual EER schemas are matched against each other in a top-down fashion, and conflicts as they are noted are assigned to sub-categories of the schematic division. When all possible schematic conflicts have been classified, a more speculative analysis of possible data-level conflicts is conducted, to determine potential problems. The remainder of this chapter presents the classification framework using examples resulting from the analysis of the ADMI, OPINS, and IMAPMIS databases. References to the assumed detailed data definitions, which are provided for extracted attribute subsets in Appendices G through I for ADMI, OPINS and IMAPMIS, respectively, are intended to be complete enough so that immediate cross checking is not required. Analysis of the appendices will reveal many potential conflicts not explicitly shown below. Chapter VI offers some potential resolution strategies for resolving the semantic conflicts between the candidate databases, and completes the integration of a global EER schema which would be used to guide the formulation of queries against the complete body of data.

1. Schematic Level Conflicts

As mentioned above, this type of conflict arises from the conceptual arrangement and definition of the databases. Since all three databases have been represented in an equivalent EER form, the process of identifying these disparities is simplified. Top-down analysis of the individual database schemas yields three subcategories of schema level conflict: entity conflicts; attribute conflicts; and entity-attribute conflicts. Entity level conflicts occur between equivalent entities. Attribute level conflicts specify discrepancies among like attributes. Entity-Attribute level conflicts concern differing organization of data, such as representing the same information as an attribute in one case, and as an entity in another.

Each subcategory will be detailed in order, with examples from the three databases under discussion.

a. Entity Level Conflicts

Entity level conflicts occur when like real world entities have differing names (synonyms), or differing entities have identical names (homonyms). Entity structures as represented by the database schema may also conflict. A third entity level conflict occurs when relationship constraints between entities differs across two or more schemas.

(1) Naming Conflicts. An example of a synonym problem is the entity COMMISSIONED_OFFICER, in the OPINS database, contrasted with the equivalent MEMBER-OFR in IMAPMIS. Both refer to instances of a particular commissioned Naval officer, but in an integrated schema, a single entity name must be specified. Similarly, IMAPMIS names a given course of college

education for a given officer EDUC, but OPINS names the same entity EDUCATION. Relatively obvious dissimilarities such as this are simple to resolve, but all kinds of complex synonym conflicts can occur in real world cases.

Homonyms are a more serious problem, because in this case, different real world entities are given the same name. Identification of homonym conflicts requires more detailed dissection of each entity, to determine its actual meaning. When completely differing concepts are captured by like-named entities, this must be rigorously checked, since an uncritical mapping of the two into a single entity in the global schema will give a meaningless result. An example of homonyms is apparent when the UNIT entity from ADMI is compared with UNIT in OPINS. In the ADMI database, UNIT refers to an instance of a military activity, such as a ship or squadron. In OPINS, however the same name is given to an entity which is actually an officer's assignment to a given billet, at a given unit. It is obvious that even though identical names are assigned to these entities, a very different semantic content is represented in the two.

(2) Entity Structure Conflicts. This is caused by overlapping or incomplete attribute sets for equivalent entities. This can arise due to failure of one database to include certain attributes captured by another because it was not considered of interest. Information concerning an entity might also be represented by the attributes of other entities in a Generalization/Specialization relationship.

An example of missing attributes is found in UNIT which in the OPINS database does not include an attribute for the unit Zip code, while UNIT in the ADMI does. The designers of the OPINS did not choose to store this particular information. Likewise, LANGUAGE in the ADMI has an attribute for listening proficiency, while LANGUAGE_SKILL in OPINS contains an attribute for writing skill. Again, this results from differing areas of informational concern when the original databases were designed.

Overlapping attributes are found in MEMBER-OFR (IMAPMIS), which does not contain the member's name, contrasted with COMMISSIONED_OFFICER (OPINS), which does. This is due to the Generalization/Specialization relationship of MEMBER-OFR to MEMBER in IMAPMIS. The member's name is represented by an attribute of MEMBER. Thus the same information is present, but at a different level of the schema. Since OPINS captures information on a more limited population than IMAPMIS, the attributes are arranged in a different manner.

(3) Constraint Conflicts. When the cardinality of relationship between two entities varies across two or more schemas, it is termed an entity constraint conflict. This is shown by the n-to-1 relation between UNIT and ACTIVE_DUTY_MILITARY_MEMBER in the ADMI, as opposed to the 1-to-n relation between COMMISSIONED_OFFICER and UNIT in the OPINS. If the structure of the entities manifesting a constraint conflict is indeed similar, this again indicates a basic semantic conflict regarding just what the databases are attempting to represent. It will be shown that in this particular instance, the constraint conflict actually results from a structure conflict because the two unit entities are dissimilar. However, constraint conflicts are independently a valid classification of semantic heterogeneity.

Another type of entity constraint conflict occurs when there is a difference in participation requirements for equivalent relationships in two databases. An example of this is given by the partial participation of **MEMBER** in the 'Has' relationship with **SCRTY-CLEAR** in the IMAPMIS. Contrast this with the total participation of **COMMISSIONED_OFFICER** in the 'Certified For' relationship with **SECURITY_REQUIREMENT** in the OPINS. This type of conflict arises from differing views of the informational domain by two groups of users. Since IMAPMIS defines the relationship of a generalization member to a security clearance (e.g., all members *may* have a security clearance), the participation constraint conflicts with that of OPINS, which models total participation (all officers *must* be certified for one and only one security requirement).

b. *Attribute Level Conflicts*

Attribute level conflicts cover the same conceptual range as the entity level. Attributes representing the same real world informational element can have differing names, or differing attributes identical names. Attribute structure conflict is analogous to entity structure conflict. Attribute constraint conflict differs from entity constraints since it is due not to relationship cardinality or participation constraints, but to differences in the attribute definition.

(1) Attribute Name Conflicts. Like entity name conflicts, this category comprises synonyms and homonyms. The reasons for this type of conflict are the same as for the entity level. Samples of attribute synonyms from the databases of interest are **DESIGNATOR** (OPINS) and **DESIG** (IMAPMIS), as well as **ORIGINAL_SOURCE_CODE** (OPINS) and **SOURCE_OF_COMMISSION** (ADMI). These both illustrate identical real world facts called by different names.

An example of homonyms is **UNIT_IDENTIFICATION_CODE** (ADMI) contrasted to **UNIT_IDENTIFICATION_CODE** (OPINS). These two identically named attributes represent different real world facts. The ADMI captures Department of Defense wide unit identification, while the same attribute in the OPINS is actually a composite attribute made up of PARENT_UIC and ACTUAL_UIC with the latter attribute corresponding to UNIT_IDENTIFICATION_CODE.

(2) Attribute Structure Conflicts. These are similar to entity structure conflicts, and arise from information being represented by an atomic attribute in one database, and the same information as either two separate attributes, or part(s) of a composite attribute in another.

Equivalent information is captured by RACE_ETHNIC in ADMI, and the two attributes RACE and ETHNIC in the OPINS. This case demonstrates a single attribute to multiple attribute structure conflict. Alternatively, the real world value of an officer's warfare designator is represented by the atomic attribute DESIG in the OPINS, while the IMAPMIS database breaks this information down into DESIG-CAT and DESIG-STAT, which themselves are part of the composite attribute DESIG.

(3) Attribute Constraint Conflicts. Unlike constraint conflict at the entity level, attribute constraint conflict occurs due to the detailed description of the attribute itself. Thus equivalent real world facts are represented by attributes which have different data definitions. This can be manifested as type clashes (e.g., character opposed to numeric), length clashes (e.g., larger or smaller number of characters in a given field), and range clashes (e.g., different allowable set of values for equivalent facts). Type conflict is quite common when dealing with databases designed for different operational implementations, while range and length conflict results more from semantic design choices.

An example of type and length clash is given by SOCIAL_SECURITY_NUMBER (ADMI), which is defined as a 4 byte packed integer, while the identical information is defined as a nine numeric integers (which can be handled as a string by modern processing techniques) for SSN (OPINS). The two Year/Month/Day attributes DATE_OF_BIRTH (ADMI) and D O B (IMAPMIS) are similarly mismatched, as the first is stored as a 3 byte packed integer, and the second as a 6 character string.

Allowable value, or range, clash, is also illustrated by the two date attributes just noted. In the IMAPMIS, the member's date of birth is defined as having a value between January 1, 1900 and December 31, 1999. An incompatible range is defined for the ADMI, since the date of birth in this database can take on any 6 digit value which corresponds to a valid date (in other words, the date is only constrained to be a date, and could represent a value outside that allowed for the same date in the IMAPMIS).

c. *Entity Attribute Conflicts*

Entity attribute conflicts arise when equivalent information is represented as an attribute of one entity in a given database, but as a separate entity in another database. This situation arises, like other structural semantic conflicts, because of conceptual design choices concerning the desired organization of information. A particular data element might be considered to be part of the aggregate data defining an entity by one design team, but the same element(s) might be considered important enough to set aside as an independent entity by another team. As in other structural conflicts, entity attribute conflicts have the effect of placing corresponding information at different levels of the schema.

An example of this is the member's security clearance information, which in the OPINS and IMAPMIS databases is represented as separate entities; **SCRTY-CLEAR** in the IMAPMIS, and **SECURITY_CLEARANCE** in the OPINS. The equivalent real world information (though less detailed) is stored by the ADMI as the composite attribute **SECURITY_INVESTIGATION**, and the atomic attribute **SECURITY_CLASSIFICATION**, both belonging to the **ACTIVE_DUTY_MILITARY_MEMBER** entity. Different views of the real world bring about these differing conceptual arrangements of the same information.

d. Completed Schematic Level Classification Framework

It should be apparent from the examples give above, that multiple simultaneous conflicts can exist at any level. Entities which have synonym conflicts can at the same time have structural and constraint mismatches. Equivalent attributes are often subject to both name, and structure/constraint conflicts. The value of the classification framework presented here is that it provides a systematic analytical tool for the identification of all schematic conflicts.

The full schematic classification portion of the framework is reiterated in Figure 17.

2. Data Level Conflicts

The full enumeration of semantic conflicts must also account for data level conflicts, even when all possible schematic conflicts have been identified and resolved. This is because even identically defined and named attributes may contain actual data values which do not agree. Data level

1. Entity Level Conflicts

   Naming Conflicts

      Synonyms (Same real world entities have same name in different dBs.)
      Homonyms (Different real world entities have same name in different dBs.)

   Structure Conflicts

      Different attribute sets
         Missing attributes
         Overlapping attributes

   Relationship Constraint conflicts

2. Attribute Level Conflicts

   Naming Conflicts

      Synonyms (Same real world entities have same name in different dBs.)
      Homonyms (Different real world entities have same name in different dBs.)

   Constraint Conflicts

      Type clash. (Equivalent real world attributes have different data type definitions in different dBs.)

      Rance clash. (Equivalent real world attributes of the same type data have different allowable range definitions in different dBs.)

   Structure Conflicts

      (Equivalent real world information is represented as a single attribute in one dB, and as either two separate, or part(s) of a composite attribute in another.)

3. Entity Attribute Level Conflicts

      (Equivalent information is represented as an attribute of an entity in one dB, and as either a separate entity, or attribute(s) of a Generalization/ Specialization entity structure in another.)

Figure 17. Framework of Conceptual Schema Level Heterogeneity

conflict can be broken down into two main types; inconsistencies, and representation conflicts. Inconsistencies refer to the case where two equivalent values for an identical instance, such as a date, or rank, do not agree when the results of a query are returned from two or more databases. Data representation conflicts cover a much more diverse spectrum of possible conflicts, arising from dissimilar expressions, dissimilar units, and dissimilar precisions. Incorporating these potential data conflicts into the classification framework completes this chapter, and results in a valuable methodological tool for complete identification of semantic heterogeneity.

a. *Inconsistencies*

Inconsistencies are easily conceptualized, and unfortunately very common, semantic conflicts. They arise from the real world process of creating, updating and maintaining databases. Different update times, human data-entry errors, or incorrect data submitted to be stored can all produce inconsistency. An inconsistency results when one database returns a given value for a specific real world element of interest, and another database returns a different value for the same element. This conflict is independent of any schematic naming or other conflict. While simply understood, and easily identified, inconsistency is the most difficult conflict to resolve. Often there is simply no other method available to reconcile an inconsistency except to go back to the original source of the data value, and determine which (if any) of the conflicting values are correct. There are other potential ways to approach the resolution problem, which will be addressed in Chapter VI, but none which are guaranteed to provide a general solution.

A simple, and obvious, example of an inconsistency is the ADMI database returning a PAY_GRADE of 4, corresponding to O4, or Lieutenant Commander, for a given commissioned officer, while the OPINS returns a value of 3 for the attribute PRESENT_GRADE, indicating a rank of Lieutenant. One of the two is incorrect, since an officer only holds one rank in the real world. Techniques for determining which value to use will be presented in Chapter VI.

b. *Data Representation Conflicts*

Data representation conflicts occur when incompatible symbols, units of measurement, or degrees of precision are used to store equivalent data elements. In general, this is due to design choices at the conceptual level caused by differing areas of interest, or levels of concern, about given real world information on the part of the database designers. One organization may wish to have very specific and precise information about an attribute of interest, while another organization might be satisfied with a general categorization of the same data. Alternatively, one design team may be accustomed to dealing with coded references to external look up tables to represent values, while another set of designers prefer to more explicitly represent values with characters. The physical implementation details of the hardware in use, and the individual processing procedures of the DBMS also influence the occurrence of data representation conflicts.

(1) Dissimilar Expressions. Dissimilar expression conflicts come about when two or more databases use the same type of data, but the values stored in the attribute have different meanings. For example, equivalent information might be represented by different character strings. An instance of this is ACTIVITY_TITLE, a character attribute which in OPINS represents the UNIT's text name, such as 'COMSURFRON THREE'. Contrast this to the attribute ACTY-LANG-NAME, also a character attribute, which IMAPMIS uses for the same information. The actual string stored in this attribute for the equivalent unit might be 'CMDR, SFC SQDRN 3'. Thus given character strings returned from the two databases, may or may not have the same meaning.

(2) Dissimilar Units. Dissimilar unit conflicts are caused by the storage of information, particularly absolute or relative measurements, in attributes with the same type, and length, and range, but with allowable values defined in different units. In the analysis of the administrative ADMI, IMAPMIS, and OPINS personnel databases, examples of this particular type of conflict are rare, since few measurements are maintained. One illustration is the **UNIV-DUR** attribute, in the IMAPMIS database, which represents a 2 character value for the length of an officer's course of instruction in weeks. This choice of units comes about through a domain analysis which indicates that the population of interest (Naval Reserve commissioned officer personnel) are likely to take shorter courses as opposed to longer courses pursued by active duty personnel. On the other hand, the **DURATION** attribute in the OPINS is also two characters (although stored as numeric integers), but represents the length of a course of instruction in months. If an attempt is made to match these two values, a dissimilar units conflict will occur. The value 20, returned from both, would mean both 20 weeks, and 20 months, respectively.

(3) Dissimilar Precisions. This type of data level conflict is due to real world information being specified at the attribute level in different degrees of precision. In other words, the same value returned from two or more databases has a different meaning because an identical range is subdivided with different levels of granularity. Consider **READING_PROFICIENCY** from the ADMI database. This 1 character attribute is constrained to the numeral values of zero through nine, with nine being defined as fluent, and zero as unacceptable, with eight gradations completing the allowable values. This provides the DMDC very precise information on the foreign language reading ability of personnel in the database. The OPINS definition for

SKILL_READ, however, while it is also a 1 character attribute, groups the allowable ten numeral range into four sub ranges, from poor, to outstanding. Obviously, although the two attributes store equivalent information in identical formats, the values from OPINS cannot be considered to give an identical level of detail as those from ADMI, since within sub ranges any value will result in one of the four broad categories being returned as a result of a query.

c. *Completed Data Level Classification Framework*

The proceeding data level conflicts will not all become apparent in the process of integrating a multidatabase from a set of heterogeneous databases. Dissimilar expressions and dissimilar precisions may or may not be identified, depending on the depth of description available to the integration effort in terms of detailed data definitions. The actual attribute definitions for the three candidate databases were not considered in this study, and the assigned data definitions have been designed to illustrate each of the possible conflict types. This is representative of the level of analysis required to identify the full range of semantic conflicts.

Unfortunately, data inconsistencies will almost certainly not become obvious, until data from global queries is returned. No level of purely conceptual analysis will be able to preclude wrong data, mismatched update times, or data entry error. Inconsistencies are included in the framework because they represent one very important type of semantic conflict, albeit one not resolvable by the conceptual integration effort.

The complete data level classification portion of the framework is reiterated in Figure 18.

1. **Inconsistencies**

   (Equivalent information returned from different dBs disagrees in value.)

2. **Data representation conflicts**

   Dissimilar Expressions
   (Equivalent information returned from different dBs is represented by incompatible values.)

   Dissimilar Units
   (Equivalent information returned from different dBs is expressed in different units.)

   Dissimilar Precisions
   (Equivalent information returned from different dBs is given to different degrees of accuracy.)

Figure 18. Framework of Data Level Heterogeneity

B. THE SEMANTIC HETEROGENEITY FRAMEWORK

The fully realized framework for classifying semantic heterogeneity can now be applied to any set of existing databases which have been transformed into equivalent EER schemas. By using a systematic approach to analysis of each equivalent set of entities, attributes, and relations, all possible semantic conflicts will be identified. Of course, for the useful integration of a set of heterogeneous databases into a global schema, these conflicts must somehow be resolved. Chapter VI addresses this issue in a general way, offering some possible solutions for each category of semantic conflict. Applying these methods of resolution, the three administrative databases under consideration will be integrated into a coherent, globally addressable EER schema. The specific instrumentalities of resolving each type of conflict, as well as a rigorous analysis of general solutions, is left to future research.

The complete framework for semantic heterogeneity is shown below as Figure 19.

Schematic Level

1. **Entity level conflicts**

Naming conflicts
Synonyms (Same real world entity has different names in different dBs.)
Homonyms (Different real world entities have same name in different dBs.)

Structure conflicts

Different attribute sets
Missing attributes
Overlapping attributes

Relationship Constraint conflicts

2. **Attribute Level Conflicts**

Naming conflicts

Synonyms (Same real world attribute has different names in different dBs.)
Homonyms (Different real world attributes have same name in different dBs.)

Constraint conflicts

Type clash. (Equivalent real world attributes have different data type definitions in different dBs.)

Range clash. (Equivalent real world attributes of the same data type have different allowable range definitions in different dBs.)

Structure conflicts

(Equivalent real world information is represented as a single attribute in one dB, and as either two separate, or part(s) of a composite attribute in another.)

3. **Entity Attribute Level Conflicts**

(Equivalent information is represented as an attribute of an entity in one dB, and as either a separate entity, or attribute(s) of a Generalization/Specialization entity structure in another.)

Figure 19. Complete framework for Semant: Heterogeneity

Data Level

1. **Inconsistencies**
   (Equivalent information returned from different dBs disagrees in value.)

2. **Data representation conflicts**
   Dissimilar Expressions
   (Equivalent information returned from different dBs is represented by incompatible values.)

   Dissimilar Units
   (Equivalent information returned from different dBs is expressed in different units.)

   Dissimilar Precisions
   (Equivalent information returned from different dBs is given to different degrees of accuracy.)

Figure 19. Complete framework for Semantic Heterogeneity (Concluded)

VI. SOLUTIONS FOR RESOLVING SEMANTIC HETEROGENEITY

A. GENERAL APPROACH

With the candidate databases for integration in equivalent schemas, and all potential sources of semantic conflict identified using the framework presented in the preceding chapter the final step can be completed. This is to consolidate them into a single global schema which can be used to guide the formulation of queries against the total set of available data. Additionally, internal design considerations of the global controller component which actually manipulates the federated database are developed during this stage of the integration process. It is during this phase that conflicts are resolved, while not losing any information.

The spectrum of possible solutions to identified semantic conflicts ranges from complete redesign of a new integrated database, to maintaining the separate databases, under some query scheme which allows them to be addressed as one. This federated database approach was described in Chapter IV, and this chapter presents in the federated database context some very general rules which can be used to resolve the conflicts noted in Chapter V. These rules apply both at the level of schema integration and data conflict resolution. The question of verifiably correct solutions to the various types of semantic conflict is a rich field of future research on integrating heterogeneous databases.

The resolution strategy presented here proceeds in two parts, forming the global schema, and dealing with data conflicts which are returned against queries. First, schema conflicts between the local schemas in the common data models are resolved, allowing them to be merged to form a global schema. This guides the user in formulating queries against the total body of data. Suggested methods for choosing the structure of the global schema are offered below for each type of conflict. Design of the global controller component is guided by these choices in the processing of queries. Second, the global controller is provided with the complete definition of the global schema, including appropriate means of mapping from the global schema to the component databases, as well as the information needed to translate, compare, and resolve the various data conflicts which will arise when data is returned from a global query.

The design of the global controller is influenced by the understanding of semantic conflicts gained during the re-engineering process. This component deals with semantic conflicts during query processing and retrieval, as well as resolving data level conflicts which occur when inconsistent data is returned for the same real world item of interest by the component databases. During querying and retrieval, the controller must know how to map from the entity and attribute names chosen for the global schema back to the actual names used in the component databases. When data is returned, the controller must have means to translate various attribute definitions into a common form, compare their values, and if possible, resolve data level conflicts before presenting the information to the user. In both these aspects, the re-engineering analyst uses detailed knowledge of the semantic conflicts existing among the component databases gained through the process described in this thesis.

The basic assumption of this chapter is that all available information is to be captured in the global schema. In other words, no attributes from any database are to be excluded if they provide data not represented elsewhere by equivalent attributes. Where data is duplicated, the rules presented below guide the choice of alternatives for inclusion in the global schema. Thus the union of attribute sets from equivalent entities is most often suggested, which ensures that missing attributes from any one database are not lost. Heuristics that identify which of several redundant overlapping attributes may be safely disregarded complete this part of the resolution process.

Another underlying assumption is that data included in the global schema should be represented in the highest level of definition or precision available. Therefore when several attributes capture equivalent information, the most precisely defined, or that which specifies the highest available degree of precision is chosen over redundant alternatives.

A final general comment on resolving semantic conflicts is that in many cases, there will simply be no other choice than to go back to the user. This is particularly true in the case of data inconsistencies as will be noted below. Re-examination of the real world data set might also be required to resolve cases of wrong data, though there are rules of thumb which can be applied with some risk of error.

The following section restates the specific semantic conflicts, by type, which were used as examples in the preceding chapter. Proposed solutions for each type of conflict are offered, with estimates of effectiveness, practicality, and certainty of correct resolution where appropriate. The completed global schema for the three officer personnel databases is presented at the end of the section on schematic conflict. This is followed by a section dealing with data level conflicts, with some considerations for the design of a global controller component for a federated database application approach to integrating them.

B. PROPOSED SOLUTIONS

The following examples duplicate, for consistency, the conflicts by type which were identified and classified in Chapter V.

1. Schematic Level Conflicts

Solutions to schematic level conflicts generally involves renaming, combining, or redefining entities and attributes in a practical way to ensure the preservation of all originally available semantic content. The global controller uses name mapping and look-up tables to allow decomposition of queries against the entity or attribute name chosen for the global schema back to the component databases. With the possible exception of constraint conflicts, the integrating designer having a clear understanding of the problem domain does not need frequent recourse to the user in resolving this level of conflict.

a. Entity Level Conflicts

Naming, structure, and constraint conflicts amongst equivalent entities is resolved by suitably renaming, and combining attribute sets to form consolidated global schema entities. Suitable look-up tables are included for the global data definition in the global controller to map between these global schema names and the existing names utilized at the component database level. Analyzing the original semantic intention of the users might be required to resolve some entity constraint conflicts.

(1) Naming Conflicts. An example of a synonym problem is the entity **COMMISSIONED_OFFICER**, in the OPINS database, contrasted with the equivalent **MEMBER-OFR** in IMAPMIS. Similarly, IMAPMIS names a given course of college education for a given officer **EDUC**, but OPINS names the same entity **EDUCATION**. For obviously equivalent entities such as these, the more fully detailed name should be chosen. Alternatively, a name from a standardized data definition which appropriately describes the global entity could be chosen.

An example of homonyms is apparent when the **UNIT** entity from ADMI is compared with **UNIT** in OPINS. In the ADMI database, UNIT refers to an instance of a military activity, such as a ship or squadron. In OPINS, however the same name is given to an entity which is actually an officer's assignment to a given billet, at a given unit. Homonym conflicts such as this usually arise because of inadequate specificity of the naming conventions employed. In this case, the **UNIT** entity in OPINS should be completely renamed as **DUTY_STATION_BILLET_ASSIGNMENT** to better reflect its intended meaning, with only those attributes which relate to an officers assignment to particular billets, current and historical. Remaining attributes of the OPINS **UNIT** entity which deal with the specific unit currently assigned will be included with the global unit entity.

It is appropriate here to mention the concept of organizationally standardized Fully Qualified Naming (FQN), on which much effort has been expended recently. FQN certainly reduces, and seeks to completely avoid, semantic conflict between data element names, and applies equally to entities and attributes. FQN specifies the semantic meaning of a data element in sufficient detail that confusion between merely similar elements is eliminated. Applied to the homonym example above, FQN would result in a name such as that suggested for OPINS, which more accurately indicates the semantic function which that entity fulfills (a record of an officer's billet assignments, and not simply information about the unit currently assigned to). Similarly, FQN for ADMI would result in a name closer to CURRENT_UNIT_ASSIGNMENT. This is a very over simplified treatment of the theory of Fully Qualified Names, and is included only to illustrate the current thrust of standardization efforts and of conventions and procedures available for resolution of this kind of conflict. Whatever approach is taken, the global controller's comprehensive definition includes mapping tables to allow decomposition of queries against global schema names back to the component databases.

(2) Entity Structure Conflicts. An example of missing attributes is found in UNIT which in the OPINS database does not include an attribute for the unit Zip code, while UNIT in the ADMI does. Likewise, LANGUAGE in the ADMI has an attribute for listening proficiency, while LANGUAGE in OPINS contains an attribute for writing skill. The resolution of missing attribute conflicts is simple. The union of attribute sets is taken for equivalent entities, which ensures that all originally available data is included in the global schema. In resolving one conflict, however, the introduction of new conflicts should be avoided. This possibility is exemplified by the technique of taking unions of different attribute sets, which solves missing attribute problems, but which may raise new overlapping conflicts.

Overlapping attributes are found in MEMBER-OFR (IMAPMIS), which does not contain the member's name, contrasted with COMMISSIONED_OFFICER (OPINS), which does. This is due to the

Generalization/Specialization relationship of MEMBER-OFR to MEMBER in IMAPMIS. The member's name is represented by an attribute of MEMBER. This is solved by decomposing COMMISSIONED_OFFICER into a Generalization/Specialization structure, segregating the appropriate attributes which apply to each part of the relationship. Choosing among remaining redundant, or overlapping attributes after this entity structure conflict is resolved requires more analysis.

Where two or more attributes from different databases represent truly equivalent data elements, the attribute with the most fully detailed name, definition, and accuracy, or a standardized data element, if available, should be chosen, and the redundant attributes excluded from the global schema. Returning to the example of overlapping attributes above, the NAME attribute from IMAPMIS would be the choice, since its specification is more semantically rich than either of the other two name attributes. The global controller needs in its detailed definition the appropriate look-up tables to match the chosen global entity to the corresponding attributes in the component databases. In this case, the more fully detailed choice is intuitive to the user, since the detailed definition of the IMAPMIS NAME attribute subsumes the definitions of the other two with no loss of meaning.

(3) Entity Constraint Conflicts. When the cardinality of relationship between two entities varies across two or more schemas, it is termed an entity constraint conflict. This is shown by the one-to-one relation between UNIT and ACTIVE_DUTY_MILITARY_MEMBER in the ADMI, as opposed to the one-to-many relation between UNIT and COMMISSIONED_OFFICER in the OPINS. As mentioned, this conflict results from the fact that UNIT in the OPINS database does not represent an equivalent entity to UNIT in the ADMI

database. This is an example of interdependency of conflicts, where one type of conflict causes another conflict of a different type. In this case, solving one (renaming the OPINS UNIT) will also resolve the other. But as seen in the case of missing attributes under entity structure, an uncritical, isolated approach to resolution of semantic conflicts can become a circular problem. The resolution of one type results in new instances of a different type of conflict. For a true cardinality or participation constraint conflict, the re-engineering analyst needs to use the constraint that reflects the actual semantics of the application area of interest. Further research into this area of resolution strategy is suggested.

b. Attribute Level Conflicts

Resolution of attribute level conflict covers the same conceptual range as the entity level. Appropriate renaming, and inclusion/elimination of missing or overlapping attributes can successfully deal with naming and structure conflicts.

(1) Attribute Name Conflicts. Like entity name conflicts, these comprise synonyms and homonyms. Samples of attribute synonyms from the databases of interest are **DESIGNATOR** (OPINS) and **DESIG** (IMAPMIS), as well as **ORIGINAL_SOURCE_CODE** (OPINS) and **SOURCE_OF_COMMISSION** (ADMI). These both illustrate identical real world facts called by different names.

An examples of homonyms is **UNIT_IDENTIFICATION_CODE** (ADMI) contrasted to **UNIT_IDENTIFICATION_CODE** (OPINS). These two identically named attributes represent different real world facts. The ADMI defines the unit identification code as an 8-character code which captures Department of Defense wide unit identification, while the same attribute in the OPINS is defined as a standard 5-character Navy Unit Identification Code (UIC). The global controller will have look-up tables to allow mapping between global and component attributes in an identical manner to that discussed in the above section on entity conflicts.

(2) Attribute Structure Conflict. This is illustrated by **RACE_ETHNIC** in ADMI, and the two attributes **RACE** and **ETHNIC** in the OPINS. This case demonstrates a single attribute to multiple attribute structure conflict. Alternatively, the real world value of an officer's warfare designator is represented by the atomic attribute **DESIG** in the OPINS, while the IMAPMIS database breaks this information down into **DESIG-CAT** and **DESIG-STAT**, which themselves are part of the composite attribute **DESIG**. The suggested resolution strategy for attribute structure conflict is to capture the available information at the finest granularity (i.e., using the largest number of attributes). If **RACE** and **ETHNIC** contain the same data as **RACE_ETHNIC**, then the global controller will decompose that query into the two atomic attributes. The same holds true for the designator information. In this way, no data is lost, and the additional flexibility to manipulate the available information in useful ways is gained over using the single combined attribute.

(3) Attribute Constraint Conflicts. An example of type clash is give by **SOCIAL_SECURITY_NUMBER** (ADMI), which is defined as a 4-byte packed integer, while the identical information is defined as a 9-character string for **SSN** (OPINS). The two Year/Month/Day attributes **DATE_OF_BIRTH** (ADMI) and **DOB** (IMAPMIS) are similarly mismatched, as the first is stored as a 3-byte packed integer, and the second as a 6-character string.

Allowable value, or range, clash, is also illustrated by the two date attributes just noted. In the IMAPMIS, the member's date of birth is defined as having a value between January 1, 1900 and December 31, 1999. An incompatible range is defined for the ADMI, since the date of birth in this database can take on any 6-digit value which corresponds to a valid date (in other words, the date is only constrained to be a date, and could represent a value outside that allowed for the same date in the IMAPMIS).

To resolve both type and range clashes, the global schema attribute is redefined to subsume the definitions of the conflicting attributes. This strategy is a very rough rule of thumb at best, since it invites instances of inconsistent data, discussed below. The global controller will have to perform the translation and comparison functions described below to deal with the potential inconsistency.

c. *Entity Attribute Conflicts*

An example of this is the member's security clearance information, which in the OPINS and IMAPMIS databases is represented as separate entities; **SCRTY-CLEAR** in the IMAPMIS, and **SECURITY_CLEARANCE** in the OPINS. The equivalent real world information (though less detailed) is stored by the ADMI as the composite attribute **SECURITY_INVESTIGATION**, and the atomic attribute **SECURITY_CLASSIFICATION**, both belonging to the **ACTIVE_DUTY_MILITARY_MEMBER** entity. Resolution of this type of semantic conflict proceeds by removing the appropriate attributes from the entity they describe in the separate database, and migrating them to the separate entity in the global schema. (This approach assumes that the global schema will always represent at a minimum the sum of independent entities from the separate databases, taking equivalence mapping into account. The global controller knows where to find the equivalent information among the component databases, even when the individual schemas present that information at different conceptual levels of organization. Thus there would be no case in which the attributes of an existing entity would be migrated to a higher order entity in the global schema. This is consistent with the basic philosophy of representing data in the global schema at the finest possible granularity.)

d. *An Integrated Global EER Schema For Three Personnel Databases*

Applying the heuristics and suggested resolution strategies listed above results in a global EER schema for the Active Duty Military Inventory, Officer Personnel Information System, and Inactive Manpower And Personnel Management Information System databases. This schema can then be used to guide the formulation of queries against the total original volume of data available across all three databases. Figure 20 shows the completed global EER schema.

2. Data Level Conflicts

Data level conflicts, which include inconsistencies, and data representation conflicts, present a much more difficult resolution problem. Often the only choice is to go back to the user, or recapture the original data from domain of interest. These conflicts only arise when data is returned from a query against the federated database. The global controller must be implemented with a capacity to deal with the extraction, conversion, comparison, and resolution of these data level conflicts. The following heuristics can be applied to the design of the global controller, but with the understanding that they are by no means assured of correct results.

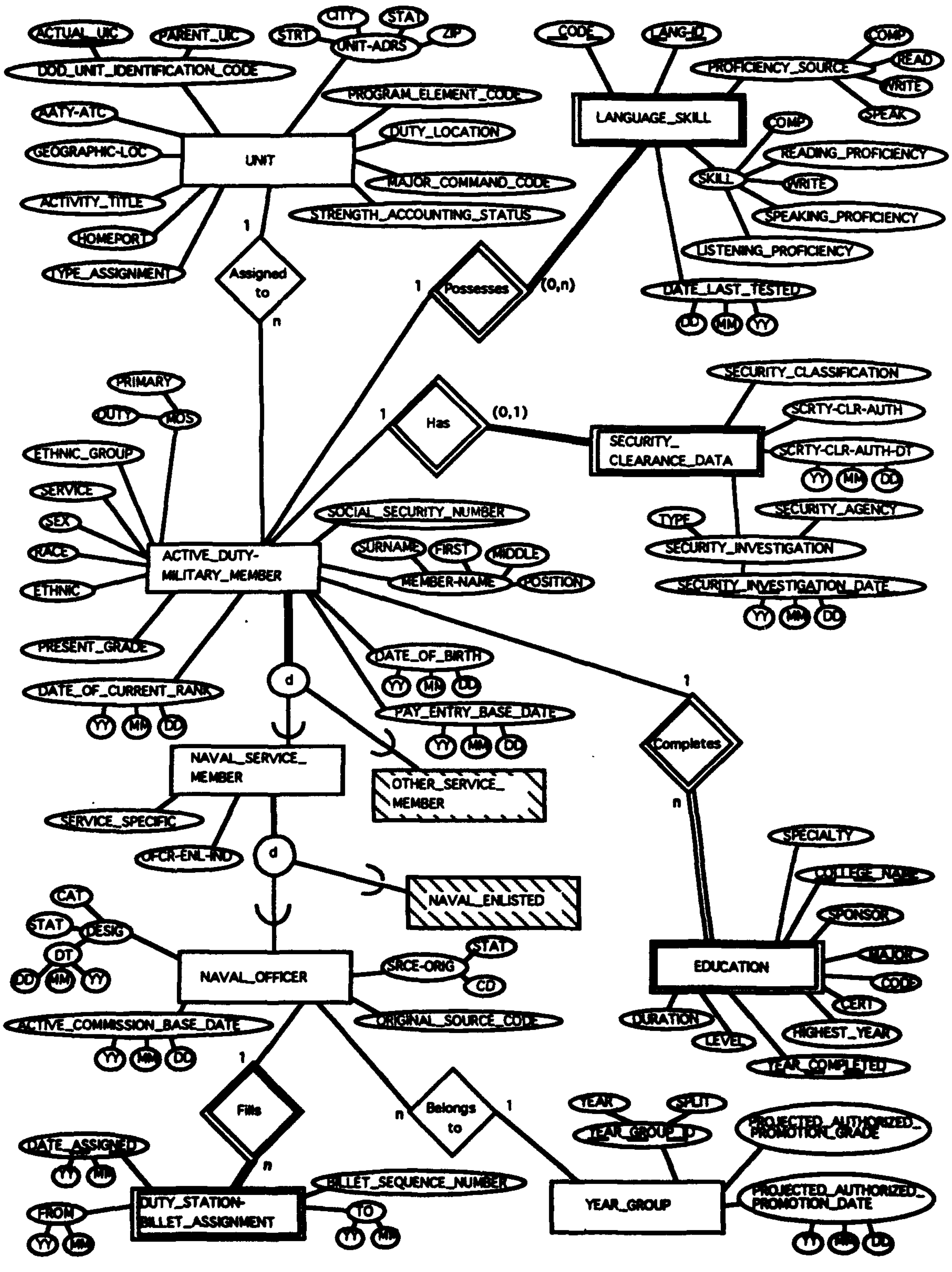


Figure 20. Integrated Global EER Schema for ADMI, OPINS and IMAPMIS

a. Inconsistencies

A simple, and obvious example of an inconsistency is the ADMI database returning a PAY_GRADE of 4, corresponding to O4, or Lieutenant Commander, for a given commissioned officer, while the OPINS returns a value of 3 for the attribute PRESENT_GRADE, indicating a rank of Lieutenant. A heuristic would be to accept the data from the database with the most recent update. This will not guarantee accuracy, but offers a simple and low effort approach. Alternatively, other data might be available to cross-verify and resolve the conflict (i.e., pay information might correspond to one rank and not another, or the DOR could be checked against years of commissioned service, to see if one rank was irrational). An important qualification of this second alternative is that it would be processing based, as opposed to a schematic resolution. This processing would be included in the detailed implementation of the global controller.

b. Data Representation Conflicts

Dissimilar expressions can often only be resolved by accepting data values from all heterogeneous databases queried by the global schema, and deciding by inspection whether the information is equivalent, and which value to accept. Alternatively, an automatic resolution might be built into the global controller. Such a solution would have to depend on large and inefficient look-up tables covering literally every conceivable expression which could represent the equivalent information of interest. This is because expression conflicts cover such a broad spectrum of possibilities, and can arise when no other classifiable conflicts are known or expected. Additionally, such a mapping scheme would necessarily be dynamic, since each new user verified instance of an equivalent, though conflicting, expression would have to be included for future reference. Dissimilar units, and dissimilar precisions admit to some general rules of thumb for resolution which are noted below.

(1) Dissimilar Expressions. An example of this is ACTIVITY_TITLE, a character attribute which in OPINS represents the UNIT's text name, such as 'COMSURFRON THREE'. Contrast this to the attribute ACTY-LANG-NAME, a character attribute, which IMAPMIS uses for the same information. The actual string stored in this attribute for the equivalent unit might be 'Cmdr, Sfc Sqdrn 3'. Bearing in mind that the more richly defined attribute was suggested above for inclusion in the global schema (in this case ACTY-LANG-NAME) the expression conflict would arise when the value from OPINS was returned and clashed with that from IMAPMIS. Further research is required to resolve this kind of conflict short of post query inspection and addition of verified equivalent representations to the look-up table, since it is a result of purely subjective choice as to appropriate content.

(2) Dissimilar Units. This is illustrated by the UNIV-DUR attribute, in the IMAPMIS database, which represents a 2-character value for the length of an officer's course of instruction in weeks. On the other hand, the DURATION attribute in the OPINS is also two characters, but represents the length of a course of instruction in months. This kind of data conflict is amenable to the FQN approach mentioned above, since one would be represented as UNIV-DUR-IN-WEEKS, with the other as DURATION_IN_MONTHS. To resolve this conflict in the context of the global controller for a federated database, each value would be retrieved, based on a (possibly) user-defined query in a given unit. The controller would accept both values, translate them into a common unit, compare them for consistency, and return the information to the user in the requested units. It is interesting to note that if the

values still conflict after translation, the conflict becomes an inconsistency, rather than a dissimilar units conflict. Additionally, in this specific case, the only time an inconsistency will arise is when the absolute values returned from the component databases match, since the original conflict is due to their difference in definition.

(3) Dissimilar Precisions. This type of data level conflict is shown by READING_PROFICIENCY from the ADMI database. This 1-character attribute is constrained to the numeral values of zero through nine, with nine being defined as excellent, and zero as unacceptable, with eight gradations completing the allowable values. The OPINS definition for SKILL_READ, however, while it is also a 1-character attribute, groups the allowable ten numeral range into four sub ranges, from zero-one meaning poor, to eight-nine meaning outstanding. In this case, the attribute definition with the finer granularity should be chosen for the global schema (to capture all available information), and during retrieval the less precise attribute values mapped onto that scale by means of a look-up table. If after this mapping, the values from the two databases still do not agree, the conflict devolves to an inconsistency, as noted above.

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Analysis of several independently developed and maintained real world databases from the same functional area shows that the expected heterogeneity does exist. Three levels of heterogeneity can be recognized; platform, DBMS, and semantic. Of these three, much effort has gone into resolving the technical problems of making a global query against databases of fundamentally different organization. Problems such as formulating a relational statement that can be processed by a CODASYL based DBMS admit to technical solutions. While this type of research addresses platform and DBMS heterogeneity, there is still an urgent need to identify and resolve semantic conflicts, or differences in the meaning of information stored in existing diverse databases.

To effectively identify and classify all types of semantic heterogeneity, data organizations must be expressed in a common schema. The Enhanced Entity Relationship model is an appropriate one for forming an integrating schema of heterogeneous databases. Because it is semantically rich, and has found wide use in initial design of databases (whatever their final implementation), it is a useful model for reverse engineering existing applications and transforming them into equivalent schemas.

By systematically comparing different schemas in the common model, the various types of semantic conflict are identifiable, and can be usefully grouped in a framework. A large part of the semantic conflicts found result from arbitrary and undisciplined application of naming conventions and data definitions during the original design. This framework represents a powerful methodological tool for the analysis of any set of heterogeneous databases which are expressed in the EER model.

The major weakness noted in this process is the difficulty of correctly capturing the original users' intentions regarding relationship constraints and cardinalities. It is felt that this is due to the fact that although relationship constraints and cardinalities are explicitly represented in an EER schema (such as one resulting from an initial top down design effort), these constraints are usually enforced at the implementation level through procedures rather than being captured in the schema itself. It is unclear that any level of database description available to the re-engineering analyst, short of a detailed source code listing of the actual application, will allow the original relationship constraints to be conceptually modeled with complete accuracy.

The process of exploring possible solutions to the various types of semantic conflict reveals that a wide spectrum of techniques apply. Some resolutions are simple, such as renaming and associated look-up tables, and provide certainty of a correct solution. Other types of semantic conflict are extremely difficult to resolve, particularly data inconsistencies. While recourse to the user, or re-examination of the real world information, will certainly deal with these problems, a more complete theoretical approach should be pursued.

B. RECOMMENDATIONS

Current Department of Defense efforts to institute Fully Qualified Naming (FQN) principles show great promise for eliminating many types of semantic conflict identified herein. FQN should be fully enforced for all new Department of Defense database applications.

FQN, however, will primarily benefit newly designed databases. There remains a need for an integrating model to support the integration of existing heterogeneous databases and the resolution of semantic conflict. This integrating model should be semantically rich enough to subsume the conceptual organizations of old and new databases. The Department of Defense should designate a suitable conceptual data model to be used in all efforts to integrate existing heterogeneous databases, and develop or procure the supporting tools to facilitate integration using the common conceptual model.

C. FUTURE RESEARCH EFFORTS

FQN will not solve the problems of semantic heterogeneity in existing databases (short of complete redesign). Therefore, further research is suggested in the area of general solutions to resolving the types of semantic conflict identified by the classification framework. In particular, interdependencies of conflicts, some of which were noted in the course of this analysis, should be more rigorously investigated. Efforts to resolve semantic conflict would benefit greatly from a framework similar to the one presented here, which could enumerate various interdependencies, and provide assured ways of resolving each, without introducing new conflicts.

Additional research is also warranted in the field of reverse engineering and the development of conceptual models for existing implementations. For example, determining relationship constraints and cardinalities from existing specifications. The ability to accurately capture this semantic content without recourse to a detailed analysis of DBMS processing algorithms would greatly enhance the usefulness of the bottom-up integration strategy suggested by this thesis.

APPENDIX A
ADMI SPECIFICATIONS

FILE FORMAT SHEET APRIL 1992

FILE NAME: ACTIVE DUTY MILITARY MASTER AND LOSS EDIT ACTDM.EDTXX.FYYMM
RECORD LENGTH: 275 X 12925 ACTDT.EDTXX.FYYMM

*See page 4 for data elements in the file, but not listed on this layout (service specific & obsolete codes) ACTDL.EDTXX.FYYMM
XX = OF, EN or CM

COL.	DESCRIPTION	COL.	DESCRIPTION
1.		51.	Y
2.	SOCIAL SECURITY NUMBER	52.	M ETS DATE
3.		53.	Y
4.		54.	M DATE CURRENT RANK
5.	TOTAL ACTIVE FEDERAL	55.	Y DATE OF LATEST ENLISTMENT (ENL)/
6.	MILITARY SERVICE	56.	M DATE OF ENTRY TO OFFICER RANKS (OFF)
7.	DOD PRIMARY OCCUPATION GROUP	57.	COMPONENT
8.		58.	YEAR OF ACTIVE DUTY SERVICE
9.	DOD DUTY OCCUPATION GROUP	59.	MONTHS IN GRADE
10.		60.	CHAR SRV (ENL LOSS)/ENL SRV CAT (OFF)
11.	EDUCATION	61.	SELECTIVE REENLISTMENT BONUS MULTIPLIER (ENL)
12.	AFQT % (ENL)	62.	(8807+)/FLYING STAT (OFF)
13.	PAY GRADE	63.*	
14.	HOME OF RECORD:STATE OR COUTRY	64.*	REENLISTMENT ELIGIBILITY (ENL LOSS)*
15.	Y	65.	Y
16.	M DATE OF BIRTH	66.	M PAY ENTRY BASE DATE
17.	D	67.	D
18.	SERVICE	68.	DUTY LOCATION:STATE OR COUNTRY (8809+)
19.	RACE	69.*	
20.	SOURCE OF COMMISSION (OFF)	70.*	
21.	EDUCATIONAL CERTIFICATION (8807+)	71.*	UNIT IDENTIFICATION CODE*
22.	MARITAL STATUS	72.*	
23.	NUMBER OF DEPENDENTS	73.*	
24.	HIGHEST YEAR OF EDUCATION (8807+)	74.*	
25.	ETHNIC GROUP	75.*	
26.	RACE ETHNIC	76.*	
27.	SEX	77.	SPANISH SURNAME FLAG
28.	TEST FORM VERS (ENL)*/YR COM SRV(OFF)	78.*	
29.	DOD SECONDARY OCCUPATION CODE	79.*	
30.		80.*	DUTY MOS (7810+)*
31.	MENTAL CAT(ENL)/A.F. COMP CAT OFF (8109+)	81.*	
32.	AGE AT ENTRY	82.*	
33.	AGE AT SEP (LOSS)/CURRENT AGE (MAST)	83.*	
34. *		84.*	
35. *		85.*	
36. *	PRIMARY MOS*	86.*	
37. *		87.*	PROGRAM ELEMENT CODE
38. *		88.*	
39. *		89.*	(SEE DOCUMENTATION FOR AVAILABILITY
40. *		90.*	INFORMATION)
41. *		91.*	
42. *	SEPARATION PROGRAM DESIGNATOR (LOSS)*	92.*	
43. *		93.*	UNIT ZIP CODE (7910+)*
44.	INTERSERVICE SEPARATION CODE (LOSS)	94.*	
45.	Y DATE OF SEPARATION (LOSS)	95.*	
46.	M SOFT ETS DATE (ENLISTED MASTER)	96.*	
47.	D ETS OF MINIMUM SERVICE REQ (OFF MAS)	97.*	SERVICE SPECIFIC - SEE DOCUMENTATION
48.	Y	98.*	
49.	M BASIC ACTICE SERVICE DATE	99.*	
50.	D	100.	GAIN/LOSS CODE

* Asterisked data items are character format

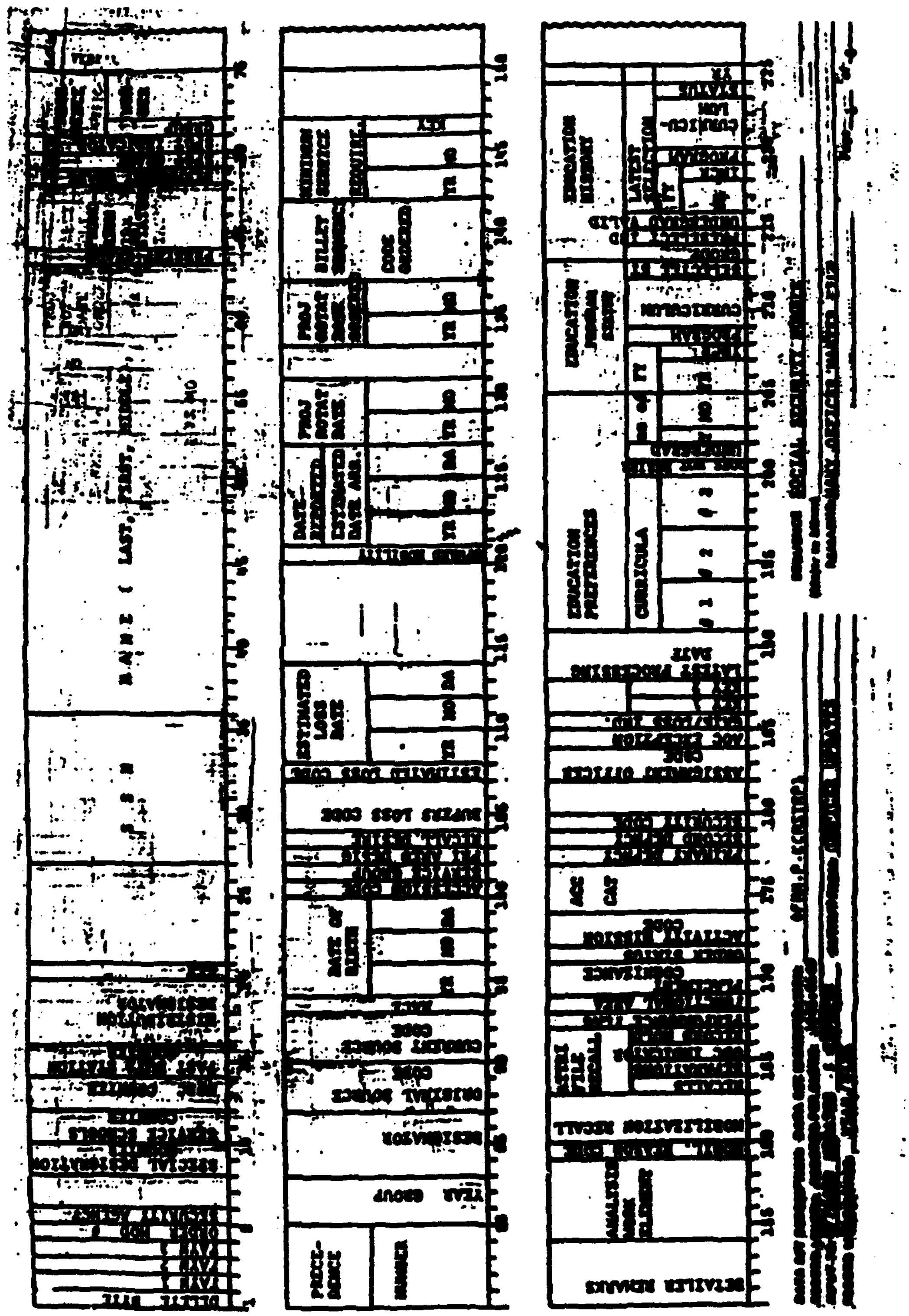
FILE NAME: ACTIVE DUTY MASTER/GAIN/LOSS EDIT FILE FORMAT CONTINUATION SHEET
MEPCOM DATA (BYTES 104-143, 145, 147-148, AND 151-160) NOT AVAILABLE ON OFFICER FILES OR FILES INCLUDING GAIN RECORDS

COL.	DESCRIPTION	COL.	DESCRIPTION
101.*		151.	AGE AT ENTRY (8712+)
102.*	HOME OF RECORDS ZIP CODE*	152.	PAY GRADE AT ENTRY (8712+)
103.*	(ENLISTED)./FOR OFFICERS-SEE	153.	EDUCATIONAL DESIGNATOR AT ENTRY (8712+)
104.*	DOCUMENTATION	154.	MEPS STATION (8712+)
105.*		155.	YOUTH PROGRAM (8712+)
106.	HIGHEST YEAR OF EDUCATION	156.	HEIGHT AT ENTRY (8712+)
107.	MARITAL/DEPENDENTS STATUS	157.	WEIGHT AT ENTRY (8712+)
108.	TEST FORM NUMBER	158.	
109.	BONUS OPTION	159.	PULHES (8712+)
110.	ENLISTMENT OPTION	160.	
111.	01	161.	SECURITY CLASSIFICATION ([illegible]+)
112.	02	162.	SECURITY INVESTIGATION ([illegible]+)
113.	03	163.	Y DATE OF COMPLETION OF
114.	04	164.	M SECURITY INVESTIGATION ([illegible]+)
115.	05	165.*	
116.	06	166.*	LANGUAGE 1 - IDENTITY ([illegible]+)*
117.	07	167.	LANG 1 - LISTENING PROFICIENCY
118.	08 ASVAB RAW SCORES	168.	LANG 1 - READING PROFICIENCY
119.	09	169.	LANG 1 - SPEAKING PROFICIENCY
120.	10	170.	LANG 1 - PROFICIENCY SOURCE
121.	11	171.	Y
122.	12	172.	M LANG 1 - DATE LAST TESTED
123.	13	173.*	
124.	14	174.*	LANGUAGE 2 - IDENTITY ([illegible]+)*
125.	15	175.	LANG 2 - LISTENING PROFICIENCY
126.	16	176.	LANG 2 - READING PROFICIENCY
127.	SERVICE	177.	LANG 2 - SPEAKING PROFICIENCY
128.	PRIOR SERVICE	178.	LANG 2 - PROFICIENCY SOURCE
129.	WAIVER CODE	179.	Y
130.	Y	180.	M LANG 2 - DATE LAST TESTED
131.	M DATE OF ENTRY	181.*	
132.	D	182.*	LANGUAGE 3 - IDENTITY ([illegible]+)*
133.	TERM OF ENLISTMENT	183.	DEFENSE LANGUAGE APTITUDE BATTERY ([illegible]+)
134.*		184.	SERVICE OF MILITARY SPOUSE (8712+)
135.*		185.	
136.*	ENLISTMENT OPTION/*	186.	SSN OF MILITARY SPOUSE (8712+)
137.*	ENLISTMENT PROGRAM	187.	
138.*		188.	
139.*		189.	PERSONNEL CLASS OF SPOUSE (8712+)
140.*		190.	
141.*	TRAINING MOS*	191.	MAJOR COMMAND CODE
142.*		192.	
143.*		193.	STRENGTH ACCOUNTING STATUS (8807+)
144.	AFQT % (ORGINAL MASTER)	194.*	
145.	MEPCOM RECORD IDENTIFICATION	195.*	
146.	RENORM FLAG	196.*	SECONDARY MOS*
147.	AFQT % (1980 METRICS)	197.*	
148.	MENTAL CATEGORY (1980 METRICS)	198.*	
149.		199.*	
150.	FILE DATE (2-BYTE BINARY)	200.*	

FILE NAME: ACTIVE DUTY MASTER/GAIN/LOSS EDIT FILE FORMAT CONTINUATION SHEET

COL.	DESCRIPTION	COL.	DESCRIPTION
201. *		251.	CONSTRUCTED CHARACTER OF SERVICE
202. *		252.*	
203. *		253.*	NAVY SECURITY INVES. TYPE (9101+)*
204. *		254.	
205. *		255.	
206. *		256.	
207. *		257.	STANDARD SCORCES (8906+)
208. *		258.	
209. *		259.	
210. *	NAME	260.	ENLISTED/TEST FORM 38+
211. *		261.	
212. *		262.	
213. *		263.	
214. *		264.	
215. *		265.	STOP-LOSS FLAG (8812+)
216. *		266.*	
217. *		267.*	
218. *		268.*	
219. *		269.*	A.F. CONTROL AFSC (9109+)*
220. *		270.*	
221. *		271.*	
222. *		272.*	
223. *		273.	
224. *		274.	
225. *		275.	
226. *		276.	
227. *		277.	
228.	SSN VERIFICATION CODE	278.	
229.	Y DATE OF INITIAL ENTRY	279.	
230.	M TO MILITARY SERVICE	280.	
231. *	SERVICE SPECIFIC*	281.	
232. *		282.	
233. *	ARMY - ARMY LOCATION CODE (8809+)*	283.	
234. *	M.C. - PAY STATUS (8812+)*	284.	
235. *	A.F. - 233-236 - INSTALLATION (8912+)*	285.	
236. *	A.F. - 237 - FUNCTIONAL CAT. (9109+)*	286.	
237. *		287.	
238.	Y	288.	
239.	M DATE ARRIVED AT STATION	289.	
240.	D (AF ONLY) (8812+)	290.	
241.		291.	
242.		292.	
243.		293.	
244.		294.	
245.	COMPOSITE SCORCES (8906+)	295.	
246.		296.	
247.		297.	
248.		298.	
249.		299.	
250.		300.	

APPENDIX B
OPINS SPECIFICATIONS



EDUCATION HIST. CODE | SERVICE COLLEGE | EFFECTIVE DATE OF PAY GRADE | PEBD PAY ENTRY BASE DATE | ADSD ACTIVE DUTY BASE DATE | A C D B ACTIVE COMMISSION DATE | DOAD DATE OF GAIN ACTIVE DUTY INITIAL | DOAD FAR-GAIN CURRENT | DATE OF FIRST COMMISSION | DOR DATE OF RANK | DEPT PROMOTION DATE OF RANK | SECTION

FC | JUNIOR | SENIOR | CURRENT DATE OF GAIN TO ACTIVE DUTY

YR | MO | COLLEGE | COLLEGE | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR

226 30 35 40 45 50 55 60 65 70 75 80 85 90 95 300

INVESTIGATION DATE | FIRST YEAR ELIGIBLE TO RETIRE (TIME) | PCN PROCESS CONTROL NUMBER | SUBMARINE SERVICE GENERAL DATE | SUBMARINE SERVICE DATE | AVIA DATE HEAVIER THAN AIR PILOT DESIGNATION DATE | NFO DATE NAVAL FLIGHT OFFICER DESIGNATION DATE | SUBMARINE QUALIFICATION DATE | SERVICE DATE YR | PROFESSIONAL SERVICE DATE | DEPENDENTS ON STATION (OVERSEAS) DATE ARRIVED | ELD PROJECTED LIST SERVICE DATE | OCCUPATION CODE | SEPARATION REASON | DESIGNATION LETTER BACKGROUND | PREVIOUS PROJECTED ROTATION DATE | AVIA CONT PAY HIST 3 DATE

MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | DAY | YEAR | MONTH | YEAR | MONTH | YEAR | MONTH | YEAR | MONTH | DAY | YR | MO | YR | MO | DAY | YR | MO | YR | MO | YEAR | MONTH

301 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75

ORDNANCE SCREEN RESULTS | ACTIVITY CATEGORY | CLAIMANT CODE | PERMANENT ASSIGNMENT CODE | DATE ASSIGNED PRES. BILLET | PRIMARY DUTY (BILLET TITLE) | BILLET SEQUENCE CODE | COLLATERAL DUTIES | DOD LOSS CODE SPD | ACTUAL LOSS DATE FY | STD INDICATOR | SUBSPECIALTY CODE 3 | EDUCATION EXECUTION CODE | LANGUAGE EXECUTION CODE | PREVIOUS

YEAR | MONTH | YEAR | MONTH | DAY | BRANCH | TPCH YR

376 80 85 90 95 400 405 10 15 20 25 30 35 40 45 453

DATA SET DESCRIPTION: DATA SET IDENTIFICATION: ______
RECORD LENGTH: RECORDS PER BLOCK: ______
INPUT TO: ______ OUTPUT FROM: ______
RECORD ARRANGEMENT: ______

SEQUENCE (Major to Minor) ______
REMARKS: ______

Page 3 of 8

MILITARY SERVICE		DATE OF RANK W2			DATE OF RANK ENSIGN			DATE OF RANK LT(JG)			DATE OF RANK LT			DATE OF RANK LCDR			DATE OF RANK CDR			DATE OF RANK CAPT			DATE OF RANK FLAG			DESIGNATOR CHANGE HISTORY						GRADUATE RECORD EXAM	
																											DESIGNATOR HISTORY 1			DESIGNATOR HISTORY 2				
MONTHS	HIGH GRADE	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DESIGNATOR	YR	MO	DESIGNATOR	VERBAL	QUANTITATIVE

531 · 55 · 60 · 65 · 70 · 75 · 80 · 85 · 90 · 95 · 500 · 505 · 10 · 15 · 20 · 525

UNREST. STRENGTH LOSS IND.	ETHNIC		EDUCATION PREFERENCE (ADJUSTED)			EDUCATION # 1							EDUCATION # 2							GRADUATE RECORD EXAM BUSINESS					NES INDICATOR	AVIA CONT PAY HIST 4			NP DES ALT
																				TEST DATE						# YRS OBLIGATED	DATE		
	R	E	PREF 1	PREF 2	PREF 3	COLLEGE NAME	DURATION	YR COMPLETED	SPONSOR	LEVEL	MAJOR	SPECIALTY	COLLEGE NAME	DURATION	YR COMPLETED	SPONSOR	LEVEL	MAJOR	SPECIALTY	YR	MO	VERBAL SCORE	QUANTITATIVE SCORE	TOTAL SCORE			YR	MO	

526 · 30 · 35 · 40 · 45 · 50 · 55 · 60 · 65 · 70 · 75 · 80 · 85 · 90 · 95 · 600

SUB-SPECIALTY CODE 1	SUB-SPECIALTY CODE 2		NUCLEAR INCENTIVE PAY									ASI EFFECTIVE DATE			SUBSPECIALTY UTILIZATION HISTORY								ACADEMIC PROFILE CODE	PRECEDENCE RECORD IND.	HEALTH PROFESSIONAL														
			NUCLEAR COMMISSION DATE			STATUS INDICATOR	NUCLEAR SERVICE CONTROL DATE			NUC SUB ACCEL IND															HEALTH PROFESSIONAL PAY ENTRY DATE			INTERN START DATE			INTERN STOP DATE			BOARD CERTIFICATION START DATE			BOARD CERTIFICATION STOP DATE	
			YR	MO	DAY		YR	MO	DAY		YR	MO	DAY	1	2	3	4	5	6	7	8			YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO	DAY	YR	MO		

601 · 605 · 610 · 15 · 20 · 25 · 30 · 35 · 40 · 45 · 50 · 55 · 60 · 65 · 70 · 675

DATA SET DESCRIPTION: DATA SET ABBREVIATION: ____________
RECORD LENGTH: RECORDS PER BLOCK: ____________
INPUT TO: ____________ OUTPUT FROM: ____________
RECORD MODE/ORDER: ____________

SEQUENCE (Major to Minor): ____________
REMARKS: ____________

Page 3 of 8

MO	DAY	PROJECTED AUTHORIZED PROMOTION DATE	FITREP INDICATOR	HEALTH PROFESSIONAL RESIDENCY START DATE YR / MO / DA	HEALTH PROFESSIONAL RESIDENCY STOP DATE YR / MO / DA		PROJECTED AUTH PROMO CHANGE	GI ACCT/OBL CODE	ACCEPT/ DECLINE DATE YR / MO / DA	RESERVE ELIG STATUS	RECOVERY STATUS	RESERVE DATE OF EXECUTION YR / MO / DA	GI BILL: RESERVE MGIB ELIGIBILITY START YR / MO / DA	GI BILL: RESERVE MGIB ELIGIBILITY STOP YR / MO / DA	DATE INITIAL ENTRY ACTIVE DUTY YR / MO / DA	RESERVE OAP GI BILL	RESERVE MONTHS REMAINING	DATE DECLINED PREVIOUS ACTIVE MGIB YR / MO / DA		GRE DATE YR / MO

676 80 85 90 95 700 705 10 15 20 25 30 35 40 45 750

GRADUATE RECORD EXAM VERB	QUANT	ANALY	DUTY STATION EFFECTIVE DATE YR / MO / DA	PAY ENTRY BASE DATE IND	DATE INDICATOR	DATE CARD IND	AB TYPE	A B DATE YR / MO / DA	BB TYPE	B B DATE YR / MO / DA	BB TYPE	E B DATE YR / MO / DA	SERVICE SCHOOL 1 CODE / DATE COMPLETED YR / MO / WEEKS DUR.	SERVICE SCHOOL 2 CODE / DATE COMPLETED YR / MO / WEEKS DUR.	SERVICE SCHOOL 3 CODE / DATE COMPLETED YR / MO / WEEKS DUR.	SERVICE SCHOOL 4 CODE / DATE COMPLETED YR / MO

751 755 60 65 70 75 80 85 90 95 800 805 10 15 20 825

WEEKS DUR	SERVICE SCHOOL 5 CODE / DATE COMPLETED YR / MO / WEEKS DUR	QUALIFICATION 1 NOBC CODE / NOBC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	QUALIFICATION 2 NOBC CODE / NOBC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	QUALIFICATION 3 NOBC CODE / CORC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	QUALIFICATION 4 NOBC CODE / NOBC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	QUALIFICATION 5 NOBC CODE / NOBC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	QUALIFICATION 6 NOBC CODE / NOBC STATION CODE / NAVAL OFFICER BILLET CLASS. NDS / NOBC KEY	NOBC CODE

826 30 35 40 45 50 55 60 65 70 75 80 85 90 95 900

DATA SET DESCRIPTION: DATA SET IDENTIFICATION: ______
RECORD LENGTH: RECORDS PER BLOCK: ______
INPUT TO: ______ OUTPUT FROM: ______
RECORD NAME/NUMBER: ______

SEQUENCE (Major to Minor) ______
REMARKS: ______

Page 4 of 8

QUALIFICATION			PAST DUTY STATION (CURRENT)						PAST DUTY STATION 2							
7	NOBC	NAVAL OFFICER BILLET CLASS	FROM	TO	ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT		FROM	TO	ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT	DEPLOYMENT DURATION	FROM
	SEA		YR MO	YR MO					YR MO	YR MO					YR

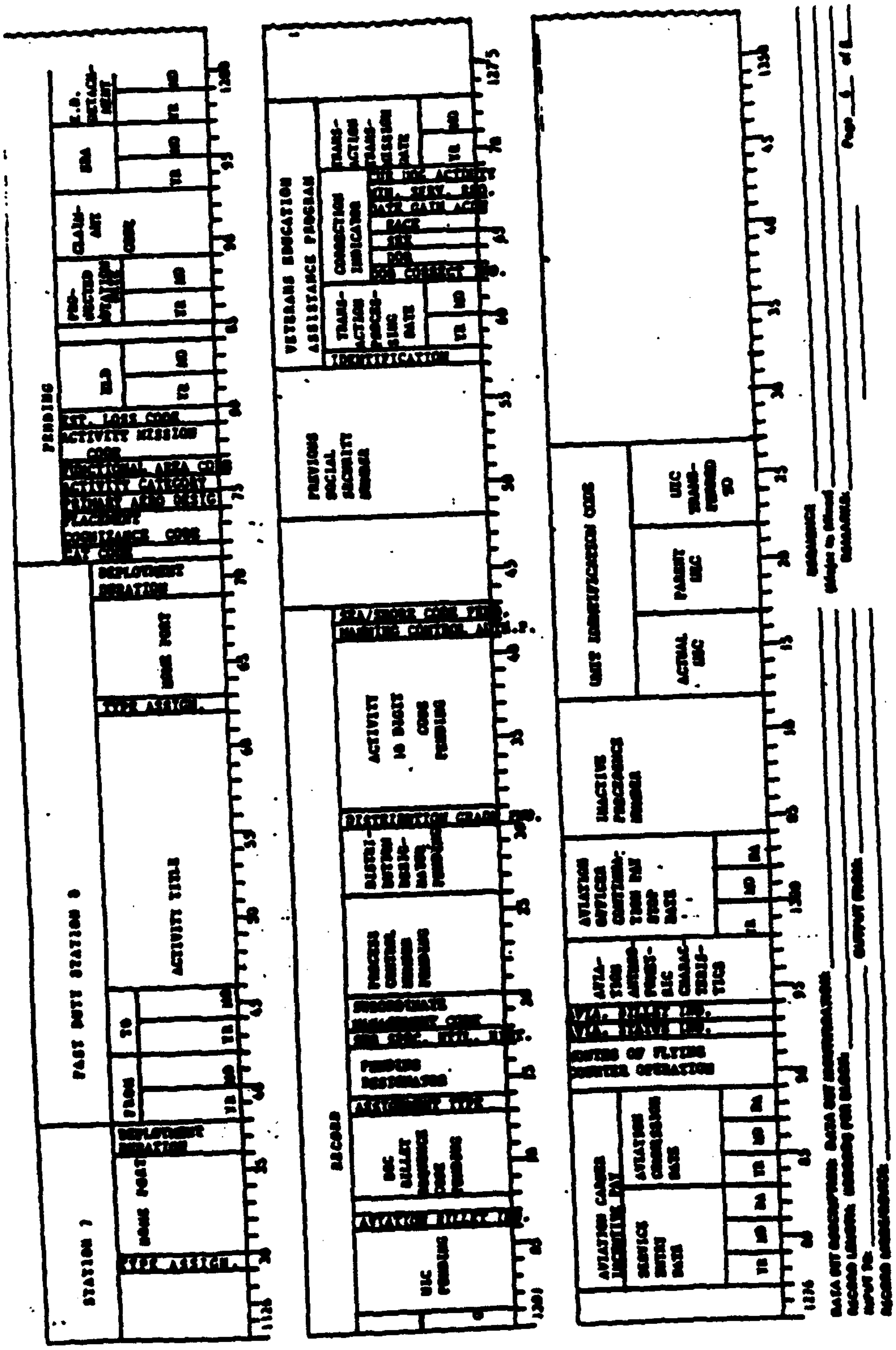
PAST DUTY STATION 3					PAST DUTY STATION 4					PAST	
TO	ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT	DEPLOYMENT DURATION	FROM	TO	ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT	DEPLOYMENT DURATION	FROM	TO
MO YR MO					YR MO	YR MO					YR MO	YR MO

DUTY STATION 5				PAST DUTY STATION 6						PAST DUTY		
ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT	DEPLOYMENT DURATION	FROM	TO	ACTIVITY TITLE	TYPE ASSIGN.	HOME PORT	DEPLOYMENT DURATION	FROM	TO	ACTIVITY TITLE
				YR MO	YR MO					YR MO	YR MO	

DATA SET DESCRIPTION: DATA SET ORGANIZATION: ____________
RECORD LENGTH: RECORDS PER BLOCK: ____________
INPUT TO: ____________ OUTPUT FROM: ____________
RECORD LOCATION: ____________

SEQUENCE (Major to Minor) ____________
REMARKS: ____________

Page 5 of 8



STATION 7
TYPE ASSIGN.
HOME PORT
DEPLOYMENT DURATION
PAST DUTY STATION 8
FROM
TO
YR
MO
ACTIVITY TITLE
PENDING
EXT. LOSS CODE
ACTIVITY MISSION CODE
FUNCTIONAL AREA CODE
ACTIVITY CATEGORY
PRIMARY AREA DESIG
COGNIZANCE CODE
PAY CODE
ELD
PROJECTED ROTATION DATE
CLAIMANT CODE
SRA
S.D. DETACHMENT
RECORD
UIC PENDING
AVIATION BILLET IND.
BSC BILLET SEQUENCE CODE PENDING
ASSIGNMENT TYPE
PENDING DESIGNATOR
PROCESS CONTROL NUMBER PENDING
DISTRIBUTION DESIGNATOR PENDING
DISTRIBUTION GRADE IND.
ACTIVITY 10 DIGIT CODE PENDING
SEA/SHORE CODE TYPE
MANNING CONTROL AUTH.
PREVIOUS SOCIAL SECURITY NUMBER
VETERANS EDUCATION ASSISTANCE PROGRAM
IDENTIFICATION
TRANSACTION PROCESSING DATE
CORRECTION INDICATOR
RACE
SEX
DOB
TRANSACTION TRANSMISSION DATE
AVIATION CAREER INCENTIVE PAY
SERVICE ENTRY DATE
AVIATION COMMISSION DATE
MONTHS OF FLYING
AVIA. BILLET IND.
AVIA. STATUS IND.
AVIATION AVIATOR CHARACTERISTICS
AVIATION OFFICER CONTINUATION PAY STOP DATE
DA
INACTIVE PRECEDENCE NUMBER
UNIT IDENTIFICATION CODE
ACTUAL UIC
PARENT UIC
UIC TRANSFERRED TO
1126
1201
1276
1200
1275
1350
DATA SET DESCRIPTION:
DATA SET IDENTIFICATION:
RECORD LENGTH:
INPUT TO:
OUTPUT FROM:
SEQUENCE
(Major to Minor)
REMARKS:
Page 6 of 8

EDA: YR | MO

ACTIVITY 10 DIGIT CODE

SEA/ SHORE CODE

MANNING CONTROL AUTH.

ORD. STATUS

ADDITIONAL DUTY RECORD: UIC | BILLET SEQUENCE CODE BSC | DATE REPORTED (YR | MO | DAY)

PREVIOUS UIC

YRS OBLIGATION

AVIA CONT PAY HIST 1: DATE (YR | MO)

135 55 60 65 70 75 80 85 90 95 1400 05 10 15 20 1425

YRS OBLIGATION

AVIA CONT PAY HIST 2: DATE (YR | MO)

AQD COUNTER

ADDITIONAL QUALIFICATION DESIGNATION

AQD 1: ADDITIONAL QUALIFICATION DESIG. CODE | AQD DATE (YR)

AQD 2: YR | CODE

AQD 3: YR | CODE

AQD 4: YR | CODE

AQD 5: YR | CODE

AQD 6: YR | CODE

AQD 7: YR | CODE

AQD 8: YR | CODE

AQD 9: YR | CODE

AQD 10: YR | CODE

AQD 11: YR | CODE

AQD 12: YR | CODE

FIRST LAN-: CODE | SKILL (COMP. | READ | SPEAK | WRITE) | MD-

C | R | S | W | C

1426 30 35 40 45 50 55 60 65 70 75 80 85 90 95 1500

GRADE: THRU (R | S | W) | PROF. YR.

SECOND LANGUAGE: CODE | SKILL (C | R | S | W) | METHOD (C | R | S | W) | PROF. YR

GAIN/LOSS IND. STRENGTH IND.

TAC # 1: JUL DATE (Y D D D) | SOURCE | TAC CODE | ERR CODE

TAC # 2: JUL DATE (Y D D D) | SOURCE | TAC CODE | ERR CODE

TAC # 3: JUL DATE (Y D D D) | SOURCE | TAC CODE | ERR CODE

TAC # 4: JUL DATE (Y D D D) | SOURCE | TAC CODE | ERR CODE

TAC # 5: JUL DATE (Y D D D) | SOURCE | TAC CODE | ERR CODE

Y D

1501 05 10 15 20 25 30 35 40 45 50 55 60 65 70 1575

DATA SET DESCRIPTION: DATA SET IDENTIFICATION: ______

RECORD LENGTH: RECORDS PER BLOCK: ______

INPUT TO: ______ OUTPUT FROM: ______

RECORD MODE/DEVICE: ______

SEQUENCE ______

(Major to Minor) ______

REMARKS: ______

Page 2 of 8

TAC #6				TAC #7				TAC #8				TAC #9				TAC #10				AVIATION OFFICER CONTINUATION PAY START DATE			ADPT / VES DESIGNATION	END OF FILE
JUL DATE	SOURCE	TAC CODE	ERR CODE	JUL DATE	SOURCE	TAC CODE	ERR CODE	JUL DATE	SOURCE	TAC CODE	ERR CODE	JUL DATE	SOURCE	TAC CODE	ERR CODE	JUL DATE	SOURCE	TAC CODE	ERR CODE	YR	MO	DA		
DD				YDDD				YDDD				YDDD				YDDD								

ONLY ON HISTORY FILE	HISTORY FILE OVERLAY OF ACTIVITY ID		
	8 - DIGIT ACTIVITY CODE		GEOGRAPHIC LOCATION CODE
	BUIC	AMC	

DATA SET DESCRIPTION: DATA SET IDENTIFICATION: ______
RECORD LENGTH: RECORDS PER BLOCK: ______
INPUT TO: ______ OUTPUT FROM: ______
RECORD MODE/DEVICE: ______

SEQUENCE (Major to Minor) ______
REMARKS: ______

Page ___ of ___

APPENDIX C
IMAPMIS SPECIFICATIONS

LOGICAL ENTITIES WITHIN THE IMAPMIS DATA BASE

KEY TO LOGICAL ENTITIES WITHIN THE IMAPMIS DATA BASE

- * ---- INDICATES PRIMARY KEY FOR A LOGICAL ENTITY.
- ● ---- INDICATES NON-UNIQUE KEY(S) AVAILABLE FOR A LOGICAL ENTITY.
- () ---- GROUP DATA ELEMENT.

AASI - AVIATION OFFICER ACIP STATUS CODES

THIS ENTITY CONTAINS ALL AUTHORIZED INACTIVE OFFICER AVIATOR ACIP STATUS VALUES AND SERVES AS A VALIDATION TABLE AS WELL AS THE PRIMARY ACCESS PATH TO ALL OCCURRENCES [illegible] STATUS VALUE.

* ACIP-STAT	[illegible]	CODE THAT REFLECTS THE CURRENT STATUS OF AN AVIATION OFFICER'S [illegible] ENTITLEMENT.
ACIP-STAT-DESC	60	CODE DESCRIPTION OR DEFINITION.
OVRDFLG-ACIP	1	FLAG TO INDICATE THAT THE KEY VALUE OF THE OCCURRENCE IS NOT CURRENTLY RECOGNIZED AS A [illegible]

AATY

THIS RELATION CONTAINS ALL OF THE UNIT IDENTIFICATION CODES OF ACTIVE UNITS [illegible]

* ACTV-UIC	5	A UNIQUE CODE ASSIGNED BY THE COMPTROLLER OF THE NAVY FOR IDENTIFICATION OF NAVAL ACTIVITIES.
O-LAST-RBSC-AATY	5	THE BILLET SEQUENCE NUMBER ASSIGNED TO THE LAST OFFICER BILLET AUTHORIZED FOR AN ACTIVE UNIT.
E-LAST-RBSC-AATY	5	THE BILLET SEQUENCE NUMBER ASSIGNED TO THE LAST ENLISTED BILLET AUTHORIZED FOR AN ACTIVE UNIT.
TEN-DGT-ACTY-CD	10	A TEN DIGIT ID NUMBER ASSIGNED TO A NAVAL ACTIVITY THAT SPECIFIES TYPE, UNIT WITHIN TYPE, AND COMPONENT WITHIN UNIT.
ACTY-PHASE-DT	(4)	THE YEAR AND MONTH THAT A PROJECTED ACTIVITY WILL BE ACTIVATED ON THE NMDAS DATA BASE.
ACTY-PHASE-DT-YR	2	CALENDAR YEAR THAT A PROJECTED ACTIVITY WILL BE ACTIVATED.
ACTY-PHASE-DT-MO	2	MONTH THAT A PROJECTED ACTIVITY WILL BE ACTIVTED.
ACTY-LNG-NM	26	A 26 CHARACTER FIELD THAT PROVIDES AN ACTIVITY'S OFFICIAL TITLE.
GEOGRAPHIC-LOC	8	AN 8 POSITION CODE THAT IDENTIFIES THE GEOGRAPHIC LOCATION OF NAVY ACTIVITIES.
MOB-LOCATOR	2	A 2 CHARACTER CODE WHICH IDENTIFIES VARIOUS MOBILIZATION SCENARIO PLANS IN WHICH THE AUIC OF A BILLET HAS AN INTEREST.
RESOURCE-SPONSOR	3	A CODE IDENTIFYING THE PROGRAM ELEMENT SPONSOR OF AN ACTIVITY.
SUB-MGMT-CMD	2	A CODE DENOTING THE COMMAND OR ACTIVITY IMMEDIATELY SUBORDINANT TO THE CLAIMANT (FORMERLY KNOWN AS TYPE COMMANDER CODES).
MNPWER-CLMNT-CD	(4)	GROUP - A CODE IDENTIFYING THE MANPOWER.
MNPWER-MAJ-CLMNT	2	FIRST TWO POSITIONS OF MANPOWER CLAIMANT CODE IDENTIFIES THE MAJOR MANPOWER CLAIMANT OF AN ACTIVITY.
MNPWER-SUB-CLMNT	2	LAST TWO POSITIONS OF MANPOWER CLAIMANT CODE IDENTIFIES THE SUB MANPOWER CLAIMANT OF AN ACTIVITY.
MARP-CD	4	A 4 DIGIT CODE ASSIGNED BY CNO FOR INTERNAL USE IN THE IDENTIFICATION OF ACTIVITIES IN THE MANPOWER REQUIREMENTS PLAN (MARP) BY FUNCTIONAL GROUPINGS ACCORDING TO ASSIGNED

		MISSION.
PARENT-UIC	5	THE UNIT IDENTIFICATION CODE WHICH IDENTIFIES THE PARENT UNIT OF AN ACTIVITY. CLAIMANT FOR AN ACTIVITY.
DPPC	2	DEFENSE PLAN PROGRAM CODE A CODE CONCERNING CERTAIN FINANCIAL ASPECTS OF MILITARY ORGANIZATIONS.
PROG-ELEMENT-CD	8	A CODE USED IN THE PLANNING, PROGRAMMING, AND BUDGETING SYSTEM.
UNIT-ADRS	(59)	THE MAILING ADDRESS OF AN ACTIVE UNIT.
UNIT-ADRS-STRT	30	A FIELD INDICATING THE ACTIVITY STREET ADDRESS.
UNIT-ADRS-CITY	18	A FIELD INDICATING THE ACTIVITY CITY ADDRESS.
UNIT-ADRS-STAT	2	A CODE INDICATING THE STATE OF THE ACTIVITY MAILING ADDRESS.
UNIT-ADRS-ZIP	9	THE ZIP+4 CODE PORTION OF THE ACTIVITY ADDRESS.
AATY-PLANG-ADDR	50	ACTIVE ACTIVITY PLAIN LANGUAGE ADDRESS
AATY-BA-TRANS-DT	(6)	ACTIVE ACTIVITY BILLET TRANSACTION DATE
AATY-BA-TRANS-YR	2	
AATY-BA-TRANS-MO	2	
AATY-BA-TRANS-DA	2	
AATY-SNAME	16	ABBREVIATED ACTIVE ACTIVITY NAME
AATY-PASS-UIC	5	ACTIVE ACTIVITY PASS UIC
AATY-MCA	1	ACTIVE ACTIVITY MANPOWER CONTROL AUTHORITY
AATY-ATC	2	ACTIVE ACTIVITY AREA TYPE CODE
AATY-HPC-ENL	1	ACTIVE ACTIVITY HOMEPORT CODE ENLISTED

********** ABCD - ABCD CODES **********

A CODE WHICH REFLECTS THE PERFORMANCE OF RESERVE OFFICERS DURING THE PAST FISCAL YEAR; BASED ON ANNIVERSERY DATE.

ABCD-STAT ABCD-STATUS	1	CODE DENOTING NUMBER OF RETIREMENT POINTS BASED ON ACTIVE STATUS

		AUTHORIZATION AT M+1 MONTH.(ZONED DEC)
TOT-M2	3	TOTAL PIM REQUIREMENTS FOR THIS AUTHORIZATION AT M+2 MONTHS.(ZONED DEC)
TOT-M3	3	TOTAL PIM REQUIREMENTS FOR THIS AUTHORIZATION AT M+3 MONTHS.(ZONED DEC)

********** MOST - MOB ORDER STATUS FILE **********

THIS ENTITY CONTAINS THE AUTHORIZED MOB ORDER STATUS CODES AND IS USED FOR EDIT PURPOSES.

* MOB-ORD-STA-MOST	1	CODE INDICATING THE MEMBER'S ORDER STATUS PRIOR TO MOBILIZATION. AFTER MOBILIZATION, IT IDENTIFIES A MEMBER'S REPORTING AND/OR INDUCTION STATUS.
MOB-ORD-STA-DESC	60	CODE DESCRIPTION OF DEFINITION.
OVRDFLG-MOST	1	FLAG TO INDICATE THAT THE KEY VALUE OF THE OCCURRENCE IS NOT CURRENTLY RECOGNIZED AS A 'LEGITIMATE' CODE. USED TO MAINTAIN KEYED ACCESS TO 'OLD' DATA ENTITIES BUT PREVENT THE USE OF THE VALUE IN 'NEW' DATA ENTRIES.

********** MSSN - MEMBER DATA - BASIC PERSONAL DATA **********

* SSN SOCIAL-SECURITY-NUMBER	9	MEMBER'S CURRENT SSN.
COCC-MXRF	9	ALSO PRIMARY KEY OF COCC.
ETHN-MXRF	1	ALSO PRIMARY KEY OF ETHN.
[illegible]	[illegible]	ALSO PRIMARY KEY OF PMRI.
[illegible]	[illegible]	IDENTIFIES THE TYPE OF NAVAL RESERVE ACTIVITY [illegible]
RACE-CD-MXRF RACE-MXRF	1	ALSO PRIMARY KEY OF RACE.
SEXC-MXRF	1	ALSO PRIMARY KEY OF SEXC.
[illegible]	[illegible]	ALSO PRIMARY KEY OF STRC.
ADDR-MXRF ADDR-ZIP-MXRF	5	ALSO PRIMARY KEY OF ADDR.

ABCD-STAT-MSSN ABCD-STATUS-MSSN	1	CODE DENOTING NUMBER OF RETIREMENT POINTS.
ABCD-CD-FY	1	
ACD-IND ACDUTRA-CMPLTN-IND	(3)	GROUP NAME -- FISCAL YEAR A MEMBER COMPLETED ACDUTRA AND STATUS OF ACDUTRA REQUIREMENT.
ACD-FY ACDUTRA-CMPLTN-FY	2	THE FISCAL YEAR A MEMBER COMPLETED HIS ACDUTRA, WAS WAIVED ACDUTRA REQUIREMENT OR WAS DETERMINED THAT ACDUTRA NOT REQUIRED.
ACD-STAT-MSSN ACDUTRA-CMPLTN-STATUS-MSSN	1	MEMBER'S ACDUTRA COMPLETION STATUS.
ACTV-LOSS-DT ACTL-ACTV-LOSS-DT	(6)	GROUP NAME -- DATE INDIVIDUAL LAST RELEASED FROM ACTIVE DUTY, EXCLUDING ACDUTRA (YYMMDD).
ACTV-LOSS-DT-YR ACTL-ACTV-LOSS-DT-YR	2	YEAR
ACTV-LOSS-DT-MO ACTL-ACTV-LOSS-DT-MO	2	MONTH
ACTV-LOSS-DT-DA ACTL-ACTV-LOSS-DT-DA	2	DAY
ADRS-STAT-MSSN ADRS-STATUS-CODE-MSSN	1	STATUS OF MEMBER'S MAIL ADDRESS.
ADRS-STREET	30	MEMBER'S STREET ADDRESS OR BOX NUMBER.
ADRS-DT-STAT ADRS-DT-STATUS	(4)	GROUP NAME -- YEAR AND MONTH THE ADDRESS OR ADDRESS-STATUS-CODE WAS CHANGED (YYMM).
ADRS-DT-STAT-YR ADRS-DT-STATUS-YR	2	YEAR
ADRS-DT-STAT-MO ADRS-DT-STATUS-MO	2	MONTH
ADRS-REL-AUTH ADRS-RELEASE-AUTH	1	CODE INDICATING RETIRED MEMBER'S AUTHORIZATION FOR RELEASE OF ADDRESS OR CANCELLATION OF THAT AUTHORIZATION.
ADRS-ZIP4	4	FOUR ADDED DIGITS TO ORIGINAL ZIP CODE WHICH ENABLES THE POSTAL SERVICE TO SORT MAIL TO A SPECIFIC CARRIER.
ANVRSRY-DT ANVRSY-DT	(4)	GROUP NAME -- DATE FOR CALCULATION OF TOTAL CREDITABLE SERVICE.
ANVRSRY-DT-MO [illegible]	2	MONTH
[illegible]		
ANVRSRY-DT-DA ANVRSY-DT-DA	2	DAY

MORST-MSSN	(2)	GROUP NAME -- MORST INFORMATION
MORST-ASSIGNMENT	1	CODE WHICH INDICATES WHICH AUTHORITY ESTABLISHED A MOBILIZATION ASSIGNMENT OR EXEMPTION. BLANK - UNASSIGNED A - CNRF MOBILIZED OR EXEMPTED B - NRPC MOBILIZED OR EXEMPTED
MORST-REASON	1	A CODE INDICATING THE MEMBER'S MOB ORDER STATUS AS TO WHETHER THE MEMBER WAS MOBILIZED OR WHY THE MEMBER WAS EXEMPTED. SEE MOST - MOBILIZATION ORDER STATUS TABLE.
CASP CAREER-SEA-PAY	(6)	GROUP NAME -- TOTAL YEARS, MONTHS AND DAYS THE MEMBER HAS ACCRUED FOR CAREER SEA PAY.
CASP-YRS CAREER-SEA-PAY-YRS	2	YEARS
CASP-MOS CAREER-SEA-PAY-MOS	2	MONTHS
CASP-DAYS CAREER-SEA-PAY-DAYS	2	DAYS
CHRCTR-SVC-MSSN	1	MEMBER'S CHARACTER OF SERVICE CODE. FOR ENLISTED MEMBERS, CODE APPLIES TO THOSE IN STRENGTH-CODE '8', '9', '0'.
CITIZ-MSSN	(2)	GROUP NAME --
CITIZ-STAT CITIZ-STATUS	1	MEMBER'S UNITED STATES CITIZENSHIP STATUS.
CITIZ-SRCE	1	SOURCE OF MEMBER'S U.S. CITIZENSHIP.
COUNTRY-OF-BIRTH	2	COUNTRY A MEMBER WAS BORN IN. ALSO PRIMARY KEY OF COBC.
DEPN-CUST-STA	1	CODE REFLECTING THE CUSTODY STATUS OF A SINGLE MBRS DEPENDENTS. A - SERVICE MEMBER HAS CUSTODY B - FORMER SPOUSE OR GUARDIAN HAS CUSTODY OR ENTITY. WHEN MEMBER IS NOT SINGLE OR HAS NO CHILDREN.
DEPEND-CD-MSSN DEPENDENCY-CD-MSSN	(2)	GROUP NAME -- CODES INDICATING NUMBER OF MEMBER'S PRIMARY AND SECONDARY DEPENDENTS.
DEPEND-PRIM DEPENDENCY-PRIM	1	CODE REFLECTING THE QUANTITY AND TYPE OF A MEMBER'S PRIMARY DEPENDENTS.
DEPEND-SEC DEPENDENCY-SEC	1	CODE REFLECTING THE QUANTITY AND TYPE OF A MEMBER'S SECONDARY DEPENDENTS.

UNCLASSIFIED

DIEAD	6	THE DATE AN INDIVIDUAL INITIALLY ENTERS ACTIVE DUTY. THIS DOES NOT INCLUDE "IADT" AND "ACDUTRA". FOR MEMBERS WITHOUT PRIOR SERVICES, RESERVE OR LOST TIME, THIS IS THE DATE THE INDIVIDUAL REPORTED TO "BOOT CAMP" OR ADSD. THIS IS REQUIRED FOR ALL NEW ACCESSIONS AND EVENTUALLY FOR ALL PERSONNEL.
DIEMS	6	THE DATE (YYMMDD) AN INDIVIDUAL WAS FIRST APPOINTED OR ENLISTED IN ANY MILITARY SERVICE OF THE UNITED STATES (ACTIVE OR RESERVE).
DIERF	6	THE DATE (YYMMDD) AN INDIVIDUAL AFFILIATES OR ENLISTS IN ANY RESERVE COMPONENT (NON-EAD) FOR THE FIRST TIME. THIS DOES NOT INCLUDE DELAYED ENTRY PROGRAM (DEP) OR PROFESSIONAL APPOINTMENT PROGRAM.
DOB	(6)	GROUP NAME -- DATE MEMBER WAS BORN. (YYMMDD) NOTE -- SOME MEMBER'S HAVE INCOMPLETE DATA E.G. (290000) (350000)
DOB-YR	2	YEAR
DOB-MO	2	MONTH
DOB-DA	2	DAY
IPENT-TYPE-MSSN IPAY-ENT-TYPE-MSSN	1	CODE INDICATING TYPE OF INCENTIVE PAY MEMBER IS ENTITLED TO.
IPENT-DT IPAY-ENT-DT	(4)	GROUP NAME -- DATE ENTITLEMENT STARTED.
IPENT-DT-YR IPAY-ENT-YR	2	YEAR
IPENT-DT-MO IPAY-ENT-MO	2	MONTH
IPENT-MO-TOT IPAY-ENT-MO-TOT	2	TOTAL MONTHS OF ENTITLEMENT.
IRAD-MSSN	4	CODE DENOTING THE APPROPRIATE READINESS LEVEL AND ATTAINMENT OF EACH INDIVIDUAL IN RELATION TO THE MEMBER'S MOBILIZATION ASSIGNMENT.
LAST-PHYS-DT	(4)	GROUP NAME -- DATE MEMBER COMPLETED LAST MEDICAL PHYSICAL EXAMINATION. (YYMM)
LAST-PHYS-YR	2	YEAR
LAST-PHYS-MO	2	MONTH

LAST-TAC-DT	(6)	GROUP NAME -- DATE UPDATED EACH TIME DATA HELD ON MEMBER IS MODIFIED.(YYMMDD)
LAST-TAC-YR	2	YEAR
LAST-TAC-MO	2	MONTH
LAST-TAC-DA	2	DAY
PURGE-TAC-DT	(4)	GROUP NAME -- TRANSACTION DATE OF LOSS USED FOR PURGE PURPOSES.
PURGE-TAC-YR	2	YEAR
PURGE-TAC-MO	2	MONTH
LSL-PAYMENT LUMP-SUM-LEAVE-PAYMENT	2	THE NUMBER OF ACCRUED LEAVE DAYS FOR WHICH A MEMBER HAS BEEN PAID SINCE 10 FEB 1976.
MAILING-CD-MRSN MAILING-CODE-MRSN SPEC-MAILING-CD-MRSN	1	CODE INDICATING WHICH MAILINGS OF RESERVE OR RETIRED INFORMATION A MEMBER SHOULD RECEIVE.
MBR-MARITAL-STAT	1	MEMBERS LEGAL STATUS AS IT RELATES TO, FIRST, MARRIAGE. ALSO PRIMARY KEY OF MARI.
MEMBER-NAME	27	MEMBER'S FULL NAME, INCLUDING SURNAME, FIRST, AND MIDDLE NAME OR NAMES, AND SUFFIX (FAMILY POSITION) IF APPLICABLE.
NRPC-LOSS-IND NRPC-TYPE-LOSS-IND	1	TYPE LOSS INDICATOR, FOR IN HOUSE USE ONLY.
OFCR-ENL-IND	1	CODE THAT IDENTIFIES MEMBER AS OFFICER OR ENLISTED.
PEBD	(6)	GROUP NAME -- ADJUSTED DATE FROM WHICH TOTAL SERVICE FOR PAY MAY BE COMPUTED (YYMMDD).
PEBD-YR	2	YEAR
PEBD-MO	2	MONTH
PEBD-DA	2	DAY
PCNT-DSBLTY-MRSN PERCENTAGE-DSBLTY-MRSN	[illegible]	THE PERCENTAGE OF PHYSICAL DISABILITY OF THE MEMBER.
PHY-ANN-CER-DT-1	4	
PHY-ANN-CER-DT-2	4	
PHY-ANN-CER-DT-3	4	
[illegible]	[illegible]	[illegible] CODE ON MEMBERS ELIGIBLE AS [illegible] FOR MOBILIZATION. ALSO PRIMARY KEY OF PIMI

 UNCLASSIFIED

PIMMS-IND-DT	6	DATE INDICATING EFFECTIVE DATE THAT A PIMMS INDICATOR VALUE WAS ASSIGNED TO A MEMBER. ONLY USED WITH CERTAIN PIMMS INDICATORS. YYMMDD
RCSBP-HILO-OPTID RCSBP-HILO-OPTION	1	IDENTIFIES WHETHER SBP HI OR LOW OPTION WAS ELECTED BY THE SERVICE MEMBER. R - REDUCED COVERAGE, F - FULL COVERAGE, BLANK
RCSBP-MBR-LTR	1	THIS ELEMENT WILL CONTAIN EITHER A 'Y', 'N' OR 'Z' CORRESPONDING TO THE MAILING OF A LETTER TO THE MEMBER INDICATING THAT THE HIGHEST RCSBP OPTION HAS BEEN ASSIGNED BY DEFAULT.
RCSBP-OPN-SN-ENL RSCBP-OPN-SN-ENL	1	IDENTIFIES WHETHER THE SBP ENROLLMENT OCCURRED DURING AN OPEN SESSION
RCSBP-OPTION RSCBP-OPTION	1	OPTION SELECTED BY THE SERVICE MEMBER A,B,C,X
RCSBP-SPOUS-LTR RSCBP-SPOUS-LTR	1	WHETHER A LETTER WAS SENT TO THE SERVICE MEMBERS SPOUSE INFORMING OF THE ELECTION SELECTED. L OR BLANK
RCSBP-TYPE-CVRGE RSCBP-TYPE-CVRGE	1	TYPE OF SBP COVERAGE ELECTED
RES-RET-ELIG-IND	1	CODE INDICATING IF MEMBER HAS BEEN NOTIFIED OF ELIGIBILITY FOR RETIRED PAY AT AGE 60. (RSRT)
RESFIRST-EXCPT RESFIRST-EXCEPTION	1	INDICATES RESFIRST TAC ERROR DETECTED AND FLAGS RECORD FOR SUBSEQUENT PROCESSING.
SAM-IND-MSSN	1	SEA, AIR, MARINE INDICATOR.
SAT-FED-SVC	(8)	GROUP NAME -- TOTAL OF A MEMBER'S FEDERAL SERVICE QUALIFYING FOR RESERVE RETIREMENT. (YYMMDD)
SAT-FED-SVC-YRS	2	YEARS
SAT-FED-SVC-MOS	3	MONTHS
SAT-FED-SVC-DAYS	3	DAYS
SCRTY-STAT SCRTY-STATUS	(8)	GROUP NAME -- MEMBER'S LAST SECURITY INVESTIGATION, INCLUDES TYPE, AGENCY AND THE DATE INVESTIGATION COMPLETED.
SCRTY-CLRNC-MSSN SCRTY-CLEARANCE-MSSN	(2)	GROUP NAME --
SCRTY-CD-MSSN SEC-INVST-TYPE	1	SECURITY CLEARANCE CODE INVESTIGATED

 UNCLASSIFIED

SCRTY-AGY-MSSN SEC-CLR-ELIG	1	INVESTIGATING AGENCY.
SCRTY-INVEST-DT SEC-INVST-DT	(6)	GROUP NAME -- DATE (YYMMDD) INVESTIGATION COMPLETED.
SCRTY-INVEST-YR	2	YEAR
SCRTY-INVEST-MO	2	MONTH
SCRTY-INVEST-DA	2	DAY
SCTY-CLEAR-DT SEC-CLR-AUTH-DT	6	DATE SECURITY CLEARANCE GRANTED
SCTY-CLEAR SEC-CLR-AUTH	1	TYPE OF SECURITY CLEARANCE HELD
SEP-DESIG-MSSN SEP-PROG-DESIG-MSSN	3	MEMBER'S SEPARATION PROGRAM DESIG CODE.
SSN-VAL	(2)	GROUP NAME -- CONTAINS CODES INDICATING THE VALIDITY OF THE SSN AND IF MILITARY WAGES HAVE BEEN CREDITED TO MEMBER'S SSN ACCOUNT. THIS DATA PROVIDED BY SSA.
SSN-COND-STAT SSN-COND-STATUS	1	SSN VALIDITY STATUS CODE.
SSN-WAGE-STAT SSN-WAGE-CREDIT-STATUS	1	SSN WAGE CREDIT STATUS CODE.
STTAX-ST	2	STATE OF LOCAL RESIDENCE OF MEMBERS FOR STATE INCOME TAX WITHHOLDING
STTAX-PCT	[illegible]	PERCENTAGE REQUESTED TO WITHHOLD (NOW REPORTED ONLY FOR THE STATE OF ARIZONA)
[illegible]	[illegible]	DOLLAR AMOUNT REQUESTED TO WITHHOLD (NOW REPORTED FOR ALL STATES EXCEPT ARIZONA)
DLAB-APT-SCORE	3	DEFENSE LANGUAGE APTITUDE BATTERY SCORE
YRS-CRED-RES-RT	2	NUMBER OF YEARS CREDITABLE TOWARD RESERVE [illegible]
[illegible]	[illegible]	TOTAL [illegible] POINTS EARNED BY THE MEMBER [illegible] RECENTLY COMPLETED RETIREMENT/ RETENTION YEAR AS OF THE END OF THE FISCAL YEAR (BASED ON INDIVIDUAL'S ANNIV YEAR)
ACR-PTS-THYR [illegible]	3	TOTAL PAID POINTS EARNED BY THE MEMBER DURING [illegible] COMPLETED RETIREMENT RETENTI[illegible] [illegible] THE FISCAL YEAR.
[illegible]	[illegible]	TOTAL [illegible] EARNED BY MEMBER CREDITABLE TOWAR RETIREMENT THROUGH THE MOST RECENTLY COMPLETED RETIREMENT/RETENTION YEAR AS OF THE END OF THE

UNCLASSIFIED

		FISCAL YEAR.
TOT-ACTV-SVC TOTAL-ACTV-SVC	(4)	GROUP NAME -- TOTAL ACTIVE DUTY (YYMM).
TOT-ACTV-SVC-YRS TOTAL-ACTV-SVC-YRS	2	YEARS
TOT-ACTV-SVC-MOS TOTAL-ACTV-SVC-MOS	2	MONTHS
TOT-DA-AFS	5	TOTAL NUMBER OF DAYS OF FEDERAL SERVICE TO INCLUDE ACTIVE DUTY (AD), TEMPORARY TOURS OF ACTIVE DUTY (TTAD) SPECIAL ACTIVE DUTY FOR TRAINING (SADT), ANNUAL TRAINING (AT), INITIAL ACTIVE DUTY FOR TRAINING (IADT), FULL TIME TRAINING DUTY (FTTD), ACTIVE DUTY FOR SUPPORT AND TEMPORARY ACTIVE DUTY (TEMAC) MAN DAYS.
TOT-PD-PT-CUR-YR	3	TOTAL ACCUMULATED PAID POINTS THIS CURRENT YEAR.
TYPE-CDR-MSSN	2	MEMBER'S ASSIGNED TYPE COMMANDER CODE. (TCOM)

************ NECC - ENLISTED NEC CODES **
*
****

* NEC-CD	4	NAVY ENLISTED CLASSIFICATION CODE.
PERS-IND	1	
NEC-SEQ-CD	1	PRIORITIZES NECS FOR ASSIGNMENT (1-4)
NEC-PG-RNG	(2)	VALID PAYGRADE RANGE TO WHICH AN NEC CAN BE ASSIGNED.(LOW,HIGH)
NEC-LO-PG	1	LOWER PAYGRADE OF RANGE
NEC-HI-PG	1	HIGHER PAYGRADE OF RANGE
NEC-STATUS	1	STATUS OF NEC (CURRENT/FUTURE/HISTORY)
NEC-RESTRICT	1	RESTRICTED TO MALE ONLY (Y/N)
NEC-DESC	40	CODE DESCRIPTION OR DEFINITION.
OVRDFLD-NECC OVER-RIDE-FLAG-NECC	1	FLAG TO INDICATE THAT THE KEY VALUE OF THE OCCURRENCE IS NOT CURRENTLY RECOGNIZED AS A LEGITIMATE VALUE. USED TO MAINTAIN KEYED ACCESS TO OLD DATA ENTITIES BUT PREVENT THE USE OF THE VALUE IN 'NEW' DATA ENTRIES.

UNCLASSIFIED

UNCLASSIFIED

PSD-CD-NRAC	2	A TWO POSITION CODE WHICH IDENTIFIES A PERSONNEL SUPPORT DETACHMENT.
NRA-POC-TEL	10	IDENTIFIES THE NRA POINT-OF-CONTACT TELEPHONE NUMBER, INCLUDING THE AREA CODE.
NRA-TYPE	1	IDENTIFIES THE NRA TYPE AS A MEGACENTER, MINICENTER, OR SATELLITE. G - MEGACENTER M - MINICENTER S - SATELLITE BLANK - NO CHANGE
NRA-SCREEN-SITE	4	IDENTIFIES THE NRA SCREENING SITE FOR THE NRAC-CODE.
NRA-NATEX-SITE	1	IDENTIFIES THE NRA AS A NATEX SITE. BLANK Y - YES N - NO
NRA-POC-NAME	27	THE NAME OF THE POINT OF CONTACT AT AN NRA.

```
**********  ODAT - OFFICER RELATED DATA  **************************************
*                                                                             *
*  THIS ENTITY CONTAINS 9 RECORD TYPES ALONG WITH THE BASE INFORMATION        *
*                                                                             *
*******************************************************************************
```

	9	PHYSICAL FIELD FOR OSSN KEY
ODATCODE	2	CODE INDICATING RECORD TYPE (AQ) ADDITIONAL QUALIFICATIONS - AQDQ (AV) AVIATION DATA - AVDT (GD) GRADE DATA - GDAT (LD) LANGUAGE DATA - LDTO (NO) NOBC DATA - NOBD (SC) SERVICE SCHOOL DATA - SCLD (SP) SPECIALTY CODE DATA - SPCD (SS) SUBSPECIALTY CODE DATA - SSCD (UN) UNIVERSITY DATA - UNDT

RECORD TYPE (AQ) ADDITIONAL QUALIFICATIONS DATA

AQD-DT	2	CODE USED TO IDENTIFY YEAR OFFICER OBTAINED ADDITIONAL-QUAL-DESIG (YY).
	--	ORDER SEQUENCING WITHIN SSN
	8	PHYSICAL FIELD WHICH IS THE KEY OF AQDC. ODATAQDC

UNCLASSIFIED

RECORD TYPE (AV) AVIATION DATA

AVIA-BIL-ASG-DT	6	THE OPERATIONAL BILLET ASSIGNMENT DATE IS THE DATE THE MEMBER WAS ASSIGNED TO A DESIGNATED OPERATIONAL FLYING SQUADRON.
ACIP-STAT-AVDT	1	ALSO PRIMARY KEY OF AASI.
AVIA-OFR-CMSN-DT	6	GROUP-NAME. A SIX POSITION CONSTRUCTIVE DATE TO INCLUDE TOTAL COMMISSIONED SERVICE ACTIVE AND/OR RESERVE.
AVIA-CUM-MOS-FLY	3	NUMBER OF MONTHS A MEMBER WAS SERVING IN AN OPERATIONAL FLYING BILLET. THIS FIELD WILL BE INCREMENTED BY 1 FOR EACH MONTH AN OFFICER CONTINUES IN AN OPERATIONAL FLYING BILLET.
AVIA-CUR-MOS-FLY	3	NUMBER OF MONTHS A MEMBER HAS SERVED IN AN OPERATIONAL FLYING BILLET. THIS FIELD WILL BE INCREMENTED BY 1 FOR EACH MONTH AN OFFICER CONTINUES IN AN OPERATIONAL FLYING BILLET.
AVIA-BIL-CAN-DT	6	THE DATE MEMBER WAS TRANSFERRED FROM A DESIGNATED OPERATIONAL FLYING SQUADRON.
AVIA-SVC-ENT-DT	6	DATE MEMBER REPORTED TO AN AVIATION ACTIVITY TO COMMENCE AN UNINTERRUPTED COURSE OF TRAINING LEADING TO AN AVIATION DESIGNATOR.

RECORD TYPE (GD) GRADE DATA

DT-GRD-EFCTV	6	EFFECTIVE DATE OF PROMOTION, SELECTION OR APPOINTMENT OF AN OFFICER TO A SPECIFIC GRAD TYPE FOR A SPECIFIC GRADE. FOR A GRD-TYPE-GD THAT EQUALS "C" (CURRENT) OR "P" (PREVIOUS), DT-GRD-EFCTV WOULD BE THE EFFECTIVE DATE OF PROMOTION. [illegible] SEQUENCED WITHIN SSN [illegible]
[illegible]	1	ALSO PRIMARY KEY OF GRAD.
GRD-TYPE-GDAT	1	ALSO PRIMARY KEY OF GTYP.
DT-SLCTN-YR	2	THE EFFECTIVE YEAR OFFICER WAS SELECTED FOR [illegible] TO A PARTICULAR GRADE.
DT-[illegible]	[illegible]	THIS [illegible] INDICATES THE DATE AN OFFICER WAS [illegible] AN INDICATED RANK (YYMMDD) FORMAT

RECORD TYPE (LD) LANGUAGE DATA

LANG-DT-TESTED	2	LAST TWO DIGITS OF YEAR IN WHICH LANGUAGE TEST WAS ADMINISTERED. (WILL EVENTUALLY BE ELIMINATED)
	--	ORDER DESCENDING WITHIN SSN
LANG-TEST-DT-OFF	4	YYMM IN WHICH LANGUAGE TEST WAS ADMINISTERED. (NEW FIELD)
	2	PHYSICAL FIELD WHICH IS THE KEY OF LANG. ODATLANG
LANG-APRSL-LDTO	(4)	GROUP-NAME CODE INDICATING MEMBER'S PROFICIENCY IN A FOREIGN LANGUAGE.
LANG-COMP-LDTO	1	CODE WHICH INDICATES A MEMBER'S PROFICIENCY IN COMPREHENDING A FOREIGN LANGUAGE.
LANG-READ-LDTO	1	CODE WHICH INDICATES A MEMBER'S PROFICIENCY IN READING A FOREIGN LANGUAGE.
LANG-SPEAK-LDTO	1	CODE WHICH INDICATES A MEMBER'S PROFICIENCY IN SPEAKING A FOREIGN LANGUAGE.
LANG-WRITE-LDTO	1	CODE INDICATING A MEMBER'S PROFICIENCY IN WRITING A FOREIGN LANGUAGE.
LANG-METH-APRSL	(4)	GROUP-NAME THE TECHNIQUE USED TO APPRAISE THE LINGUISTIC ABILITY OF AN INDIVIDUAL IN A LANGUAGE.
LANG-METH-COMP	1	CODE INDICATING THE TECHNIQUE USED TO APPRAISE COMPREHENSION ABILITY IN A FOREIGN LANGUAGE.
LANG-METH-READ	1	CODE INDICATING THE TECHNIQUE USED TO APPRAISE READING ABILITY IN A FOREIGN LANGUAGE.
LANG-METH-SPEAK	1	CODE INDICATING THE TECHNIQUE USED TO APPRAISE SPEAKING ABILITY IN A FOREIGN LANGUAGE.
LANG-METH-WRITE	1	CODE INDICATING THE TECHNIQUE USED TO APPRAISE WRITING ABILITY IN A FOREIGN LANGUAGE.

RECORD TYPE (ND) OFFICER NOBC DATA

NOBC-QUAL-DT-YR	2	QUAL DATE YEAR (YY) THAT OFFICER LEFT ACTIVITY WHERE QUALIFICATION

UNCLASSIFIED

		WAS OBTAINED OR YEAR OFFICER SUBMITTED ANNUAL QUALIFICATION FOR CIVILIAN EMPLOYMENT THAT EARNED NOBC.
	--	ORDER DESCENDING WITHIN SSN
NOBC-CD-NOBD	4	ALSO PRIMARY KEY OF NOBC.
NOBC-STN-CD-NOBD	3	CODE USED TO IDENTIFY SHIP/STATION WHERE EXPERIENCE AND QUALIFICATIONS WERE ACQUIRED.
OCCUR-TYPE-NOBD	1	THIS IS A ONE POSITION FIELD TO CATEGORIZE NOBC CODES HELD BY TYPE. P = PRIMARY, S = SECONDARY, H = HISTORY.

RECORD TYPE (SC) SERVICE SCHOOL DATA

SVC-SCHL-COMP-DT	4	YEAR AND MONTH A MEMBER SUCCESSFULLY COMPLETED A SERVICE SCHOOL(YYMM).
	--	ORDER DESCENDING WITHIN SSN
	3	PHYSICAL FIELD WHICH IS THE KEY OF SCHL.
SVC-SCHL-DURAT	2	LENGTH IN WEEKS OF A SERVICE SCHOOL MEMBER SUCESSFULLY COMPLETED.

RECORD TYPE (SP) SPECIAL CODE DATA

SPEC-CD-ASGN-DT	6	THE DATE AN OFFICER WAS ASSIGNED AN ASSOCIATED SPECIALTY
	--	ORDER DESCENDING WITHIN SSN
	1	PHYSICAL FIELD WHICH IS THE KEY OF SPEC.
SPEC-CD-TERM-YR	2	CALENDER OR FY CATEGORY TERMINATES OR IS SUBJECT TO CHANGE

RECORD TYPE (SS) SUBSPECIALTY DATA

SUBSPEC-SSDT	4	PRIMARY KEY OF SSCD.
SUBSPEC-LVL	1	IDENTIFIES THE LEVEL OR AMOUNT OF EDUCATION, TRAINING OR EXPERIENCE AN OFFICER POSSESS.

RECORD TYPE (UN) EDUCATION DATA

EDUC-ORDR-CNTRL	(4)	USED TO PRIORITIZE EDUCATION DATA BY EDUCATIONAL LEVEL PRIORITY AND DATE COMPLETED.
EDUC-LVL-PRI	2	IDENTIFIES THE RELATIVE PRIORITY OF A MEMBER'S PARTICULAR UNIVERSITY DEGREE OR EXPERIENCE. ELEMENT IS USED STRICTLY FOR INTERNAL PROCESSING TO

		SORT UNIVERSITY EDUCATION DATA IN SEQUENCE FROM HIGHEST TO LOWEST PRIORITY REGARDLESS OF DATE COMPLETED. A PRIORITY IS ASSIGNED TO EACH EDUCATION LEVEL CODE. USED IN CONJUNCTION WITH EDUC-UNIV-DT-CMP TO PROVIDE A COMPLETE SORT KEY.
	--	ORDER DESCENDING WITHIN SSN
EDUC-UNIV-DT-CMP	2	THE LAST TWO DIGITS OF YEAR DEGREE WAS AWARDED OR LAST YEAR OF ATTENDANCE.
	--	ORDER DESCENDING WITHIN SSN, EDUCATIONAL LEVEL PRIORITY
EDUC-UNIV-NAME	10	NAME OR ABBREVIATION OF NAME OF THE INSTITUTION AN OFFICER ATTENDED.
EDUC-UNIV-DUR	2	THE DURATION OF A COURSE IN MONTHS. RECORDED FOR NAVY SPONSORED POST-GRAD COURSE ONLY.
EDUC-UNIV-SPNSR	1	IDENTIFIES NAVY-SPONSORED POSTGRAD AND ADVANCED COURSES OR SPECIALIZED PROGRAMS.
EDUC-UNIV-LVL	1	CODE USED TO IDENTIFY THE HIGHEST LEVEL OF AN OFFICERS EDUCATION.
EDUC-UNIV-MAJ	2	IDENTIFIES THE MAJOR ACADEMIC FIELD
EDUC-UNIV-SPEC	2	A CODE WHICH INDICATES AN OFFICER'S SPECIALTY WITHIN A MAJOR FIELD.
CLASS-STAND-UNDT	4	AN OFFICER'S STANDING (POSITION) IN HIS/HER OR HER CLASS UPON GRADUATION. THIS DATA ONLY APPLIES TO U.S. NAVAL ACADEMY GRADUATES (I.E., EDUC-UNIV-NAME = "USNA").
CLASS-SIZE-UNDT	4	TOTAL NUMBER OF STUDENTS IN AN OFFICER'S GRADUATING CLASS. THIS DATA ONLY APPLIES TO U.S. NAVAL ACADEMY GRADUATES (I.E., EDUC-UNIV-NAME = "USNA").

********** [illegible] - OFFICER GENERAL DATA **************************************

* THIS ENTITY CONTAINS COMMON GENERAL DATA FOR EACH INACTIVE OFFICER AND IS THE "CORE" RELATION OF THE OFFICER DATA CLUSTER I.E. AN OCCURRENCE (ONE ONLY) OF OFFICER GENERAL DATA MUST EXIST BEFORE RELATIONSHIPS CAN BE ESTABLISHED TO OTHER OFFICER RELATIONS SUCH AS [illegible] DATA, [illegible] DATA, ETC. *

****

* SSN-OFR	9	THE SSN USED AS A KEY FOR THE OFFICER

SSN-GDAT SSN-UNDT SSN-SSDT SSN-LDTG SSN-NOBD SSN-AQDD SSN-SPCD SSN-AVDT SSN-SCLD		LOGICAL RELATION.
DEG2-OSSN	2	ALSO PRIMARY KEY OF DEG2.
BCO1-OSSN	1	ALSO PRIMARY KEY OF BCO1.
BASE-DT-CMSN-SVC	(6)	GROUP-NAME A SIX DIGIT DATE REFLECTING THE DATE AN OFFICER ACCEPTED HIS ORIGINAL APPOINTMENT ABOVE THE GRADE CHIEF WARRANT OFFICER.
BASE-DT-CMSN-YR	2	
BASE-DT-CMSN-MO	2	
BASE-DT-CMSN-DA	2	
BRCL-DT-OFR	(6)	GROUP-NAME EFFECTIVE DATE OF OFFICER'S CURRENT BRANCH CLASS SERVICE CODE.
BRCL-DT-OFR-YR	2	
BRCL-DT-OFR-MO	2	
BRCL-DT-OFR-DA	2	
BRCL-CD-OFR-OSSN	(2)	GROUP NAME OFFICER'S CURRENT BRANCH CLASS SERVICE CODE.
BRCL-FIRST-OFR	1	
BRCL-SECOND-OFR	1	
CONSTV-SVC-IND	(3)	GROUP-NAME CODE AND LAST TWO DIGITS OF FISCAL YEAR WHICH REFLECTS THE FISCAL YEAR FROM WHICH TOTAL COMMISSIONED SERVICE FOR ATTRITION WILL BE COMPUTED.
CONSTV-SVC-CD	1	IDENTIFIES WHETHER AN OFFICER WAS COMMISSIONED BEFORE OR AFTER JAN 19, 1961.
CONSTV-SVC-FY	2	REFLECTS THE FISCAL YEAR FROM WHICH THE TOTAL COMMISSIONED SERVICE FOR ATTRITION WILL BE COMPUTED.
DESIG-OSSN	(4)	GROUP NAME OFFICER'S CURRENT DESIGNATOR CODE.

UNCLASSIFIED

DESIG-CAT	3	FIRST THREE POSITIONS OF OFFICER'S DESIGNATOR CODE.
DESIG-STAT	1	FOURTH DIGIT OF DESIGNATOR USED TO IDENTIFY THE STATUS OF AN OFFICER.
DESIG-DT-OSSN	6	THE EFFECTIVE DATE AN OFFICER WAS ASSIGNED HIS CURRENT DESIGNATOR. (YYMMDD)
ENSIGN-PROM-IND	1	HOLD AUTOMATIC PROMOTION FROM ENSIGN TO LTJG.
GRD-LIMIT-DT	(6)	GROUP-NAME THE DATE AN OFFICER IN THE READY RESERVE WILL REACH HIS GRADE LIMIT DATE.
GRD-LIMIT-DT-YR	2	
GRD-LIMIT-DT-MO	2	
GRD-LIMIT-DT-DA	2	
HLTHPROF-PROM-DT	(6)	GROUP NAME. DATE APPLICABLE TO MEDICAL CORPS OFFICERS BELOW THE GRADE OF CAPTAIN FOR DETERMINING INITIAL APPOINTMENT RANK. DATE OF RANK AND LINEAL NUMBER ASSIGNED.
HLTHPROF-PROM-YR	2	
HLTHPROF-PROM-MO	2	
HLTHPROF-PROM-DA	2	
HONORARY-PROM	1	CODE INDICATING THIS OFFICER WAS ADVANCED IN GRADE ON THE RETIRED LIST BY REASON OF HAVING BEEN SPECIFICALLY COMMENDED FOR PERFORMANCE OF DUTY IN ACTUAL COMBAT.
IPAY-OFR-OSSN	1	TYPE OF INCENTIVE PAY.
LOSS-IND	1	INDICATION THAT AN OFFICER IS A LOSS [illegible] LOSS TO STRENGTH.
[illegible]	1	INDICATES THAT DATA PROVIDED TO IMAPMIS FROM OPINS VIA DET HAS AN ERROR. THEREFORE A NON-CRITICAL DATA ELEMENT MAY BE MISSING FROM THE DATA RECORD.
[illegible]-DT	(4)	GROUP-NAME THE YEAR AND MONTH [illegible] WAS FORWARDED TO [illegible] RECALL.
OFR-CORRES-YR	2	

UNCLASSIFIED

UNCLASSIFIED

OFR-CORRES-MD	2	
OVR-4-YR	1	CODE INDICATING OFFICER HAS OVER FOUR YEARS ACTIVE ENLISTED SERVICE IN ONE OR MORE BRANCHES OF THE ARMED SERVICE.
PHYS-RISK-OSSN	1	OFFICER'S PHYSICAL FITNESS CODE.
PREC-NMBR-INACT	(8)	GROUP-NAME AN 8 DIGIT NUMBER ASSIGNED A RESERVE OFFICER INDICATING HIS POSITION ON THE PRECEDENCE LIST OF OFFICERS ON INACTIVE DUTY.
PREC-NMBR-MAIN	6	FIRST SIX CHARACTERS OF OFFICER PRECEDENCE NUMBER.
PREC-NMBR-SUB	2	LAST TWO DIGITS OF PRECEDENCE NUMBER USED FOR INSERTIONS.
PROM-AUTH-DT	(6)	GROUP-NAME THE DATE OF ISSUANCE OF ORIGINAL APPOINTMENT. USED TO DETERMINE SENIORITY.
PROM-AUTH-DT-YR	2	
PROM-AUTH-DT-MO	2	
PROM-AUTH-DT-DA	2	
PROM-STATUS	(4)	GROUP-NAME CODE INDICATING AN OFFICERS SELECTION OR NON-SELECTION FOR PROMOTION TO NEXT HIGHER GRADE THAN HIS PRESENT GRADE.
PROM-STAT-OSSN	1	INDICATES IF AN OFFICER HAS BEEN SELECTED FOR PROMOTION, FAILED SELECTION, OR HAS HAD NORMAL PROMOTION WITHHELD.
PROM-STAT-YR	3	ONE TO THREE DIGIT CODE REPRESENTING THE FISCAL YEAR AN OFFICER WAS SELECTED OR FAILED SELECTION FOR PROMOTION.
PRSNT-STATUS-DT	(6)	DATE-OF-STATUS FOR THE OFFICER- THIS IS EITHER THE DATE HE ACQUIRED HIS PRESENT STATUS OR THE DATE THE OFFICER'S PRESENT STATUS WILL EXPIRE, DEPENDING ON HIS BRCL.
PRSNT-STATUS-YR	2	
PRSNT-STATUS-MO	2	
PRSNT-STATUS-DA	2	

UNCLASSIFIED

QUAL-RCRD-REV-YR	2	REFLECTS CALENDAR YEAR IN WHICH QUALIFICATION RECORD OF AN OFFICER WAS REVIEWED AND CODES REVISED.
RECALL-DSRB-OSSN	1	INDICATES THE DESIRABILITY OF AN OFFICER FOR RECALL PURPOSES AFTER SEPARATION FROM ACTIVE DUTY.
REINSTATEMENT-DT	(6)	GROUP-NAME EFFECTIVE DATE OF AN INDIVIDUAL'S TRANSFER FROM USMR-S1 TO READY RESERVE.
REINSTATEMENT-YR	2	
REINSTATEMENT-MO	2	
REINSTATEMENT-DA	2	
RET-ACT-CD-OSSN	(2)	GROUP-NAME CODE WHICH REFLECTS THE LEGISLATIVE ACT UNDER WHICH AN OFFICER WAS RETIRED.
RET-ACT-1	1	CODE REFLECTING THE FIRST LEGISLATIVE ACT UNDER WHICH AN OFFICER WAS RETIRED.
RET-ACT-2	1	CODE REFLECTING THE SECOND LEGISLATIVE ACT UNDER WHICH AN OFFICER WAS RETIRED.
RET-TYP-OFR-OSSN	2	CODE SHOWING OFFICER'S TYPE OF RE- TIREMENT.
RPRT-IND1-OFR	1	
RPRT-IND2-OFR	1	
[illegible]	1	
RPRT-IND4-OFR	1	
RPRT-IND5-OFR	1	
[illegible]	1	
[illegible]	1	CODE INDICATING WHETHER OR NOT A DISCHARGE [illegible] WAS BEEN PRODUCED.
SRCE-STAT-O-OSSN	(3)	GROUP NAME. OFFICER'S ORIGINAL SERVICE SOURCE AND IF THAT SERVICE WAS ACT/INACT.
[illegible]	2	A [illegible] INDICATES THE PROGRAMS UNDER WHICH AN [illegible] FIRST RECEIVED A COMMISSION.

108

UNCLASSIFIED

SOURCE-STAT-OFR	1	INDICATES PRIOR ACTIVE OR INACTIVE SERVICE.
TRNG-CAT-IND	1	CODE INDICATING THE STATUS OF RECORDS RELATIVE TO PRODUCING A TRAINING CATEGORY ASSIGNMENT LETTER.
TYPE-GAIN-CD-OFR	1	CODE FOR OFFICERS INDICATING TYPE/ REASON FOR GAIN TO MASTER FILE.
TYPE-LOSS-CD-OFR	1	CODE FOR OFFICERS INDICATING TYPE/ REASON FOR LOSS TO MASTER FILE.
UNI-MAINT-ALLOW	1	REPRESENTS TERMINAL DIGIT OF THE YEAR AN OFFICER MAY BECOME ELIGIBLE FOR UMA.
YRS-CMSND-SVC	2	NUMBER OF YEARS AN OFFICER SERVED IN A COMMISSIONED STATUS, ACTIVE AND INACTIVE.

```
**********  PDIS - PERCENT OF DISABILITY TABLE  ******************************
*                                                                            *
*  ALLOWABLE VALUES FOR PHYSICAL DISABILITY PERCENTAGES.                     *
******************************************************************************
```

* PCNT-DSBLTY	2	ALLOWABLE VALUES FOR PHYSICAL DISABILITY PERCENTAGES.
OVRDFLG-PDIS	1	FLAG TO INDICATE THAT THE KEY VALUE OF THE OCCURRENCE IS NOT CURRENTLY RECOGNIZED AS A 'LEGITIMATE' CODE. USED TO MAINTAIN KEYED [illegible]

```
**********  PHYS - OFFICER PHYSICAL RISK CLASSIFICATION CODES  ****************
*  [illegible]                                                               *
```

* PHYS-RISK	1	CODE INDICATING LEVEL OF OFFICER PHYSICAL FITNESS FOR ACTIVE DUTY.
[illegible]		[illegible] THAT THE KEY VALUE OF THE OCCURRENCE IS NOT CURRENTLY RECOGNIZED AS A 'LEGITIMATE' CODE. USED TO MAINTAIN KEYED

APPENDIX D
ADMI ENTITY STRUCTURE

Active Duty Master Inventory (ADMI)

ACTIVE_DUTY_MILITARY_MEMBER =
SOCIAL_SECURITY_NUMBER (key)
+ NAME (comp)
+ DATE_OF_BIRTH (comp)
+ SEX
+ RACE_ETHNIC
+ ETHNIC_GROUP
+ PAY_ENTRY_BASE_DATE (comp)
+ SERVICE
+ MOS (comp)
+ DATE_OF_CURRENT_RANK (comp)
+ PAYGRADE
+ SECURITY_CLASSIFICATION
+ SECURITY_INVESTIGATION (comp)
+ EDUCATION (comp)

NAVAL_SERVICE_MEMBER =
+ NAVAL_SECURITY_INVESTIGATION_TYPE
+ SERVICE_SPECIFIC

NAVAL_OFFICER =
+ YEARS_OF_COMMISSIONED_SERVICE
+ SOURCE_OF_COMMISSION

UNIT =
UNIT_IDENTIFICATION_CODE (key)
+ DUTY_LOCATION
+ UNIT_ZIP_CODE
+ MAJOR_COMMAND_CODE
+ PROGRAM_ELEMENT_CODE
+ STRENGTH_ACCOUNTING_STATUS

LANGUAGE =
+ IDENTITY (pkey)
+ DATE_LAST_TESTED (comp)
+ LISTENING_PROFICIENCY
+ SPEAKING_PROFICIENCY
+ READING_PROFICIENCY
+ PROFICIENCY_SOURCE

APPENDIX E
OPINS ENTITY STRUCTURE

Officer Personnel Information System (OPINS)

COMMISSIONED_OFFICER =
  SSN (key)
+ NAME (comp)
+ DATE_OF_BIRTH (comp)
+ SEX
+ RACE
+ ETHNIC
+ PEBD (comp)
+ ACTIVE_COMMISSION_BASE_DATE (comp)
+ DESIGNATOR
+ DOR (comp)
+ PRESENT_GRADE
+ ORIGINAL_SOURCE_CODE

UNIT =
  UNIT_IDENTIFICATION_CODE (comp) (sub-attribute is key)
+ HOMEPORT
+ TYPE_ASSIGNMENT
+ ACTIVITY_TITLE
+ BILLET_SEQUENCE_NUMBER (key)
+ DATE_ASSIGNED (comp) (key)
+ FROM (comp)
+ TO (comp)

LANGUAGE_SKILL =
+ CODE (pkey)
+ PROFICIENCY_YEAR
+ METHOD (comp)
+ SKILL (comp)

SECURITY_REQUIREMENT =
  SECURITY_CODE (pkey)
+ SECURITY_AGENCY
+ SECURITY_INVESTIGATION_DATE (comp)

YEAR_GROUP =
    YEAR_GROUP_ID (comp) (KEY)
  + PROJECTED_AUTHORIZED_PROMOTION_DATE (comp)
  + PROJECTED_AUTHORIZED_PROMOTION_GRADE

EDUCATION =
    COLLEGE_NAME (pkey)
  + YEAR_COMPLETED (pkey)
  + LEVEL
  + DURATION
  + MAJOR
  + SPECIALTY
  + SPONSOR

APPENDIX F
IMAPMIS ENTITY STRUCTURE

Inactive Manpower And Personnel Managment Information Sysytem (IMAPMIS)

MEMBER =
SSN (key)
+ MEMBER-NAME (comp)
+ DOB (comp)
+ SEXC
+ RACE
+ ETHN
+ PEBD (comp)
+ OFCR-ENL-IND

MEMBER-OFR =
+ GRD-CD
+ SRCE-ORIG (comp)
+ DT-PRMTN (comp)
+ DESIG (comp)
+ BASE-DT-CMSN-SVC (comp)

EDUC =
+ UNIV-NAME (pkey)
+ UNIV-DT-CMP (pkey)
+ UNIV-SPNSR
+ UNIV-DUR
+ UNIV-LVL
+ UNIV-MAJ
+ UNIV-SPEC

AATY =
+ ACTIV-UIC (key)
+ ACTY-LANG-NAME
+ GEOGRAPHIC-LOC
+ UNIT-ADRS (comp)
+ AATY-ATC
+ PROG-ELEMENT-CD

SCRTY-CLEAR =
+ SCRTY-INVST-DT (pkey) (comp)
+ SCRTY-AGCY
+ SCRTY-INVST-TYPE
+ SCRTY-CLR-AUTH
+ SCRTY-CLR-AUTH-DT (comp)

LANG =
+ LANG-ID (pkey)
+ LANG-APRSL (comp)
+ LANG-METH-APRSL (comp)
+ LANG-DT-TESTED (comp)

APPENDIX G
ADMI DATA DEFINITIONS

Active Duty Master Inventory ADMI

Attribute	Type	Length	Key	Range	Constraint
ACTIVE_DUTY_MILITARY_MEMBER Entity					
SOCIAL_SECURITY_ NUMBER	NI	4	Y	0-9	Mandatory
**Member's Social Security Number (in 4 byte packed numeric format).					
NAME	C	27	N	A..Z	
-LAST		15			
-FIRST		11			
-MIDDLE		1			
**Member's full name (in Last,First,MI (including ","s) format).					
DATE_OF_BIRTH	NI	3	N	0-9	
**Member's date of birth (in YYMMDD 3 byte packed numeric format).					
SEX	NI	1	N	1,2	
**Member's sex (1=Male, 2=Female).					
RACE_ETHNIC	C	1	N	C,M,A,I,H	
**Member's Race (Caucasian, African, etc.).					
ETHNIC_GROUP	C	1	N	A-Z, 0-9	
**Member's ethnic group (special code).					
PAY_ENTRY_ BASE_DATE	NI	3	N	0-9	
**Member's start date for calculation of time in service for pay purposes (in YYMMDD 3 byte packed numeric format).					
SERVICE	NI	1	N	1,2,3,4	Mandatory
**Member's service (1=Army, 2=Air Force, 3=Navy, 4=Marines). Defining attribute of SERVICE-MEMBER Specialization.					
MOS	C	14	N	A-Z, 0-9	
-PRIMARY		7			
-DUTY		7			
**Member's Military Occupational Specialty (code, both the MOS gained by training (Primary), and for the current assignment (Duty)).					

Attribute	Type	Length	Key	Range	Constraint
DATE_OF_CURRENT_ RANK	NI	2	N	0-9	
**Member's date of promotion to current pay grade (in YYMM 2 byte packed numeric format).					
PAYGRADE	NI	1	N	0-9	
**Member's current paygrade (1=E1/O1, 2=E2/O2, etc., covering grades for Naval Officers from Ensign {O1} to Admiral {O10}).					
SECURITY_ CLASSIFICATION	NI	1	N	0-9	
**Member's security clearance (0=None, 1=Classified, 2=Secret, 3=Top Secret, etc.).					
SECURITY_ INVESTIGATION	NI	3	N	0-9	
-TYPE		1			
-DATE_OF_ COMPLETION		2			
**Type of security investigation completed for member (0=None, 1=National Agency Check, 2=Background Investigation, 3=Special Background Investigation, etc.), and date on which it was completed (in YYMM 2 byte packed numeric format).					
EDUCATION	NI	3	N	0-9	
-CODE		1			
-CERT		1			
-HIGHEST_YEAR		1			
**Member's educational data, including code for college level courses (0=No, 1=Yes), certification of High School completion (0=No, 1=Yes), and highest year of schooling completed (in 1 byte packed numeric format).					
NAVAL_SERVICE_MEMBER Entity					
NAVAL_SECURITY_ INVESTIGATION_ TYPE	C	2	N	0-9	
**Special security investigation information required for Naval service members (field is null for other services).					
SERVICE_SPECIFIC	C	2	N	0-9	
**Meaning of attribute varies according to member's service.					
NAVAL_OFFICER Entity					
YEARS_OF_ COMMISSIONED_ SERVICE	NI	1	N	0-9	
**Officer's total years of commissioned military service (in 1 byte packed numeric format).					

Attribute	Type	Length	Key	Range	Constraint
SOURCE_OF_ COMMISSION	NI	1	N	0-9	

**Officer's commissioning source (0=Service Academy, 1=ROTC, 3=Officer Candidate School, etc.).

UNIT Entity

Attribute	Type	Length	Key	Range	Constraint
UNIT_ IDENTIFICATION_ CODE	C	8	Y	A-Z, 0-9	Mandatory

**Department of Defense 8 digit Unit Identification Code (includes 3 digit service/component identification, plus 5 digit std Navy UIC)

Attribute	Type	Length	Key	Range	Constraint
DUTY_LOCATION	NI	1	N	0-9	

**Unit's geographic location (0=Continental US, 1=Europe, 2=Japan, 3=Middle East, etc.).

Attribute	Type	Length	Key	Range	Constraint
UNIT_ZIP_CODE	C	5	N	0-9	

**Unit's 5 digit postal Zip code.

Attribute	Type	Length	Key	Range	Constraint
MAJOR_ COMMAND_CODE	NI	3	N	0-9	Mandatory

**Unit's assignment to major force command (121=CINCPACFLT, 333=USAREUR, 542=Sixth FLT, etc.).

Attribute	Type	Length	Key	Range	Constraint
PROGRAM_ ELEMENT_CODE	C	6	N	A-Z, 0-9	Mandatory

**Unit's budgetary funding program element code.

Attribute	Type	Length	Key	Range	Constraint
STRENGTH_ ACCOUNTING_ STATUS	NI	1	N	0, 1	Mandatory

**Unit's is required to continuously report total percentage of authorized end stregth (0=No, 1=Yes).

LANGUAGE Entity

Attribute	Type	Length	Key	Range	Constraint
IDENTITY	C	1	P	A-Z	Mandatory

**Foreign language (F=French, R=Russian, M=Mandarin, A=Arabic, etc.).

Attribute	Type	Length	Key	Range	Constraint
DATE_LAST_ TESTED	NI	2	N	0-9	

**Date on which the language proficiency was last tested (in YYMM 2 byte packed numeric format).

Attribute	Type	Length	Key	Range	Constraint
LISTENING_ PROFICIENCY	NI	1	N	0-9	

**Level of apptitude in listening comprehension for a foreign language (0=Unacceptable, 1=Very Poor, 2=Poor, 3=Below Average, 4=Average, 5=Above Average, 6=Good, 7=Very Good, 8=Excellent, 9=Fluent).

Attribute	Type	Length	Key	Range	Constraint
READING_ PROFICIENCY	NI	1	N	0-9	

**Level of apptitude in Reading comprehension for a foreign language (0=Unacceptable, 1=Very Poor, 2=Poor, 3=Below Average, 4=Average, 5=Above Average, 6=Good, 7=Very Good, 8=Excellent, 9=Fluent).

Attribute	Type	Length	Key	Range	Constraint
SPEAKING_ PROFICIENCY	NI	1	N	0-9	

**Level of apptitude in speaking comprehension for a foreign language (0=Unacceptable, 1=Very Poor, 2=Poor, 3=Below Average, 4=Average, 5=Above Average, 6=Good, 7=Very Good, 8=Excellent, 9=Fluent).

Attribute	Type	Length	Key	Range	Constraint
PROFICIENCY_ SOURCE	NI	1	N	0-9	

**Source of the proficiency ratings for Listening, Speaking and Reading (0=Assessment by supervisor on duty, 1=Local Test, 2=Formal language school, 4=Defense Language Institues, etc.).

APPENDIX H
OPINS DATA DEFINITIONS

Officer Personnel Information System OPINS

Attribute	Type	Length	Key	Range	Constraint
COMMISSIONED_OFFICER Entity					
SSN	NI	9	Y	0-9	Mandatory
**Member's Social Security Number.					
NAME	C	27	N	A..Z	
-LAST		16			
-FIRST		10			
-MIDDLE		1			
**Member's full name (in Last , First , MI format, including spaces).					
DATE_OF_ BIRTH	NI	6	N	0-9	
**Member's date of birth (in YYMMDD format).					
SEX	C	1	N	M, F	
**Member's sex.					
RACE	C	1	N	A-Z	
**Member's race (C=Caucasian, N=Negroid, H=Hispanic, etc.).					
ETHNIC	C	1	N	A-Z	
**Member's ethnic group (arbitrary code, A=North European, B=Canadian, C=East European, etc.).					
PEBD	NI	6	N	0-9	
**Member's pay entry base date (in YYMMDD format).					
ACTIVE_ COMMISSION_ BASE_DATE	NI	6	N	0-9	
**Member's starting date of commissioned service (in YYMMDD format).					
DESIGNATOR	NI	4	N	0-9	Mandatory
**Officer's warfare designator (1110=Active Duty Surface Warfare, 1115=Reserve Surface Warfare, etc.).					
DOR	NI	6	N	0-9	
**Member's date of present grade (in YYMMDD format).					

Attribute	Type	Length	Key	Range	Constraint
PRESENT_GRADE	NI	1	N	0-9	Mandatory

**Officer's current paygrade (1=O1, 2=O2, etc., covering grades from Ensign (O1) to Admiral (O10)).

ORIGINAL_ SOURCE_CODE	C	1	N	A-Z	

**Officer's original commissioning source (A=Naval Academy, R=Reserve Officer Training Corps, O=Officer Candidate School, etc.).

UNIT Entity

UNIT_ IDENTIFICATION_ CODE	NI	10	N	0-9	Mandatory
-ACTUAL_UIC		5		K	
-PARENT_UIC		5		Y	

**Navy 5 digit unit identification code for both unit assigned, and Immediate Superior in Command (ISIC) of that unit.

HOMEPORT	C	6	N	A-Z	

**Plain language name (or abbreviation) of unit's assigned homeport.

TYPE_ ASSIGNMENT	NI	1	N	0-9	

**Unit's duty type assingment (0=Sea, 1=Continental US, 2=Overseas, etc.).

ACTIVITY_TITLE	C	16	N	A-Z, 0-9	

**Unit's plain language title (or abbreviation).

BILLET_SEQUENCE_ NUMBER	NI	5	K	0-9	

**Specific duty assignment by billet number (12345=Commanding Officer, 67890=Executive Officer, etc.).

DATE_ ASSIGNED	NI	4	K	0-9	

**Date assignment was made to the specific duty billet (in YYMM format).

FROM	NI	4	N	0-9	

**Date the specific duty billet the specific duty billet assignment was assumed (in YYMM format).

TO	NI	4	N	0-9	

**Date the specific duty billet assignment was vacated (in YYMM format).

Attribute	Type	Length	Key	Range	Constraint
LANGUAGE_SKILL Entity					
CODE	NI	2	P	0-9	
Mandatory					
**Foreign language (01=Spanish, 43=French, 11=Russian, 24=Arabic {Iraqi dialect}, 52=Farsi, etc.).					
PROFICIENCY_ YEAR	NI	2	N	0-9	
**Year in which proficiency in a language was most recently tested (in YYMM format).					
METHOD	C	4	N	0-9	
-COMP		1			
-READ		1			
-WRITE		1			
-SPEAK		1			
**Method used to appraise the level of aptitude in comprehension, reading, speaking, writing, and speaking a language (0=Assessment by supervisor on duty, 1=Local Test, 2=Formal language school, 4=Defense Language Institute, etc.).					
SKILL	C	4	N	0-9	Mandatory
-COMP		1			
-READ		1			
-WRITE		1			
-SPEAK		1			
**Level of apptitude in comprehension, reading, speaking, writing, and speaking a language (0-1=Poor, 2-4=Average, 5-7=Good, 8-9=Outstanding).					
SECURITY_REQUIREMENT Entity					
SECURITY_ CODE	NI	1	P	0-9	Mandatory
**Level of security classification for which investigation requirements have been completed (0=None, 1=Classified, 2=Secret, 3=Top Secret, etc.).					
SECURITY_ AGENCY	C	6	N	A-Z	
**Agency abbreviation which completed security investigation NVINSV=Naval Investigative Sevice, DFINSV=Defense Investigative Sevice, FDBUIN=Federal Bureau of Investigation, CTINAY=Central Intelligence Agency, etc.).					
SECURITY_ INVESTIGATION_ DATE	NI	6	N	0-9	
**Date of completion of security investigation (in YYMMDD format).					

Attribute	Type	Length	Key	Range	Constraint

YEAR_GROUP Entity

YEAR_GROUP_ID	NI	3	Y	0-9	Mandatory
-YEAR		2			
-SPLIT		1			

**Promotian year group (in YY , plus {0=No split, 1=Split, lower half, 2=Split upper half} format).

PROJECTED_ AUTHORIZED- PROMOTION_ DATE	NI	6	N	0-9	

**Prospective date of promotion to next higher rank for memebers of the year group (in YYMMDD format).

PROJECTED_ AUTHORIZED- PROMOTION_ GRADE	NI	1	N	0-9	

**Prospective next rank of memebers of the year group (1=O1, 2=O2, etc., covering grades from Ensign {O1} to Admiral {O10}).

EDUCATION Entity

COLLEGE_NAME	C	10	P	A..Z	Mandatory

**Educational institution name (or abbreviation)

YEAR_ COMPLETED	NI	2	P	0-9	Mandatory

**Year in which a course of education was completed (in YY format).

LEVEL	C	1	N	A-Z	

**Level of course of education (U=Undergraduate, G=Graduate, P=Postgraduate).

DURATION	NI	2	N	0-9	

**Duration (in months) of course of education.

MAJOR	NI	2	N	0-9	

**Academic major (12=Oceanography, 43=Aeronautical Engineering, 55=Electrical Engineering, etc.).

SPECIALTY	NI	2	N	0-9	

**Naval warfare specialty associated with course of education (24=Surface Warfare, 55=Antisubmarine Warafre, 87=Anitair Warfare, etc.).

SPONSOR	NI	1	N	0-9	

**Navy organization which sponsored course of education (3=Op-03, 4=Op-04, 8=Op--8, etc.).

APPENDIX I
IMAPMIS DATA DEFINITIONS

Inactive Manpower And Personnel Managment Information Sysytem IMAPMIS

Attribute	Type	Length	Key	Range	Constraint
MEMBER Entity					
SSN	C	9	Y	0-9	Mandatory
**Member's Social Security Number.					
MEMBER-NAME	C	27	N	A..Z, 1-9	
-SURNAME		13			
-FIRST		7			
-MIDDLE		5			
-POSITION		2			
**Member's full name (includes JR., SR, 2, 3, etc).					
DOB	C	6	N	0-9	btwn 01/01/00 and 01/01/99
**Member's date of birth (in YYMMDD format).					
SEXC	C	1	N	M/F	
**Member's sex					
RACE	C	1	N	C,M,A,I,H	
**Member's Race (Caucasian, African, etc.).					
ETHN	C	1	N	0-9	
**Member's ethnic group (code).					
PEBD	C	6	N	0-9	btwn 01/01/00 and 01/01/99
**Member's Pay Entry Base date (in YYMMDD format).					
OFCR-ENL-IND	C	1	N	O,E	Mandatory
**Member's Officer/Enlisted status.					
MEMBER-OFR Entity					
GRD-CD	C	1	N	0-9	Mandatory
**Officer's present rank (O1-O9).					

Attribute	Type	Length	Key	Range	Constraint
SRCE-ORIG	C	3	N	a..Z, 0-9	
-SRCE-CD		2			
-SRCE-STAT		1			
**Officer's original commissioning source and active/reserve status (Code).					
DT-PRMTN	C	6	N	0-9	btwn 01/01/00 and 01/01/99
**Member's date of Rank (in YYMMDD format).					
DESIG	C	10	N	0-9	
-DESIG-STAT		1			
-DESIG-CAT		3			
-DESIG-DT		6			
**Officer's warfare designator (3 digit specialty, and 1 digit active/reserve ind) and date of award of designator (in YYMMDD format).					
BASE-DT-CMSN-SVC	C	6	N	0-9	btwn 01/01/00 and 01/01/99
**Member's date of commencement of commission service (in YYMMDD format).					
EDUC Entity					
UNIV-NAME	C	10	P	A..Z	Mandatory
**Educational institution name (or abbreviation)					
UNIV-DT-CMP	C	2	P	0-9	Mandatory
**Year in which a course of education was completed (in Yy format).					
UNIV-SPNSR	C	1	N	A-Z, 0-9	
**Navy organization which sponsored course of education (3=Op-03, 4=Op-04, 8=Op-8, C=Navy Comptroller, P=Bureau of Naval Personnel, R=Chief of Naval Reserve Force, etc.).					
UNIV-DUR	C	2	N	0-9	
**Duration (in weeks) of course of education.					
UNIV-LVL	C	1	N	U,G,P	
**Level of course of education (U=Undergraduate, G=Graduate, P=Postgraduate).					
UNIV-MAJ	C	2	N	0-9	
**Academic major (OC=Oceanography, AE=Aeronautical Engineering, EE=Electrical Engineering, etc.).					
UNIV-SPEC	C	2	N	0-9	
**Naval warfare specialty associated with course of education (SW=Surface Warfare, AS=Antisubmarine Warafre, AA=Anitair Warfare, etc.).					

Attribute	Type	Length	Key	Range	Constraint
AATY Entity					
ACTIV-UIC	C	5	Y	0-9	Mandatory
**Naval 5 digit Unit Identification Code					
ACTY-LANG-NAME	C	26	N	A-Z, 0-9	
**Unit's plain language title.					
GEOGRAPHIC-LOC	C	8	N	0-9	
**Unit's geographic location (code).					
UNIT-ADRS	C	59	N	A-Z. 0-9	
-UNIT-ADRS-STRT		30			
-UNIT-ADRS-CITY		18			
-UNIT-ADRS-STAT		2			
-UNIT-ADRS-ZIP		9			
**Unit's full mailing address (including 9 digit Zip code).					
AATY-ATC	C	3	N	0-9	
**Unit's area type code (Overseas, Conus, etc.).					
PROG-ELEMENT-CD	C	8	N		
**Unit's budgetary funding program element code.					
SCRTY-CLEAR Entity					
SCRTY-INVST-DT	C	6	P	0-9	btwn 01/01/00 and 01/01/99
**Date of completion of security investigation (in YYMMDD format).					
SCRTY-INVST-TYPE	C	1	N	0-9	
**Type of security investigation completed (Code).					
SCRTY-AGCY	C	1	N	0-9	
**Agency which completed security investigation (code).					
SCRTY-CLR-AUTH	C	1	N	U,C,S,T	
**Level of security classification authorized as a result of the security investigation.					
SCRTY-CLR-AUTH-DT	C	6	P	0-9	btwn 01/01/00 and 01/01/99
**Date on which security classification was authorized (in YYMMDD format).					
LANG Entity					
LANG-ID	C	2	P	0-9	Mandatory
**Foreign language (SP=Spanish, FR=French, RU=Russian, IQ=Arabic {Iraqi dialect}, FA=Farsi, etc.).					

Attribute	Type	Length	Key	Range	Constraint
LANG-APRSL	C	4	N	0-9	Mandatory
-LANG-COMP		1			
-LANG-READ		1			
-LANG-WRITE		1			
-LANG-SPEAK		1			

**Level of apptitude in comprehension, reading, speaking, writing, and speaking a language (0-1=Poor, 2-4=Average, 5-7=Good, 8-9=Outstanding).

Attribute	Type	Length	Key	Range	Constraint
LANG-METH-APRSL	C	4	N	0-9	
-LANG-METH-COMP		1			
-LANG-METH-READ		1			
-LANG-METH-WRITE		1			
-LANG-METH-SPEAK		1			

**Method used to appraise the level of aptitude in comprehension, reading, speaking, writing, and speaking a language (Code).

Attribute	Type	Length	Key	Range	Constraint
LANG-DT-TESTED	C	6	N	0-9	btwn 01/01/00 and 01/01/99

**Date on which language aptitude was appraised (in YYMMDD format).

LIST OF REFERENCES

1. Markowitz, Victor M., and Makowsky, Johann A., "Identifying Extended Entity-Relationship Object Structures in Relational Schemas," *IEEE Transactions on Software Engineering*, v. 16, no. 8, p. 777, Aug 1990

2. Kim, Myung-Joon, Lee, Won-Ung, and Derniame, "Automatic Relational Data Base Designs by Translation of the Entity-Relationship Model," *Proceedngs of the IEEE International Conference on Computers and Applications,*, p. 418, 1987

3. Elmasri, Ramez, and Navathe, S.B., *Fundamentals of Database Systems,* Benjamin/Cummings Publishing Co, pp. 37-60; 409-424, 1989

BIBLIOGRAPHY

Batini, C., Lenzerini, M., and Navathe, S. B., "A Comparative Analysis of Methodologies for Database Schema Integration," *Computing Surveys*,, vol. 18, no. 4, Association for Computing Machinery, Dec 1990

Desai, Bipin C., and Pollock, Richard, "On Schema Integration in a Heterogeneous Distributed Database Management System," *Proceedings of the IEEE International Computer Software and Applications Conference*, 1989

Elmagarmid, Ahmed K., "When Will We Have True Heterogeneous Databases?," *Proceedings of the IEEE Fall Joint Computer Conference*, 1987

Elmasri, Ramez, and Navathe, S.B., *Fundamentals of Database Systems,* Benjamin/Cummings Publishing Co, 1989

Gogolla, Martin, and Hohenstein, Uwe, "Towards a Semantic View of an Extended Entity-Relationship Model," *ACM Transactions on Database Systems*, vol. 16, no. 3, Sep 1991

Kamel, Magdi N., and Hsiao, David K., "Interoperability and Integration Issues in Heterogeneous Database Environments," *Proceedings of Database '90*, Jun 1990

Kent, William, "The Many Forms of a Single Fact," *Proceedings of the IEEE Computer Society International Conference*, 1989

Kim, Myung-Joon, Lee, Won-Ung, and Derniame, Jean-Claude, "Automatic Relational Data Base Designs by Translation of the Entity-Relationship Model," *Proceedings of the IEEE International Conference on Computers and Applications*, 1987

Kim, Won, and Seo, Jungyun, "Classifying Schematic and Data Heterogeneity in Multidatabase Systems," *Computer*, v. 42, no. 12, IEEE, Dec 1991

Krishnamurty, Ravi, Litwin, Witold, and Kent, William, "Language Features for Interoperability of Databases with Semantic Discrepancies," *Proceedings of the International Conference on Management of Data*, Association for Computing Machinery, 1991

Larson, James A., Navathe, Shamkant B., and Elmasri, Ramez, "A Theory of Attribute Equivalence in Databases with Application to Schema Integration," *IEEE Transactions on Software Engineering*, vol. 15, no. 4, Apr 1989

Markowitz, Victor M., and Makowsky, Johann A., "Identifying Extended Entity-Relationship Object Structures in Relational Schemas," *IEEE Transactions on Software Engineering*, v. 16, no. 8, Aug 1990

Olson, Philip A., "Corporate Information Management, Data Standardization," *Proceedings of Database '90*, Jun 1990

Wang, Y. Richard, and Madnick, Stuart E., "The Inter-Database Instance Identification Problem in Integrating Autonomous Systems,"*IEEE Hawaiian International Conference on System Sciences*, vol.3, 1989

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Department Chairman, Code AS Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Prof. Magdi N. Kamel AS/KA Naval Postgraduate School Monterey, CA 93943-5000	6
5.	Prof. Kishore Sengupta AS/SE Naval Postgraduate School Monterey, CA 93943-5000	1
6.	Rebecca Wade, Code DA Naval Information Sysytems Management Center Washington Navy Yard Bldg 166 Washington, D. C. 20374-5000	2
7.	LT Daniel A. Lindsey Department Head Class 126 Surface Warfare Officer School Command Newport, RI 02841-5012	2